# MATHEMATICS FOR SOFTWARE ENGINEERING

Richard Brooks & Eduard Fekete

# Mathematics for Software Engineering

**Authors:** Richard Brooks & Eduard Fekete

**Date:** May, 2025

**Version:** 2.0

# Contents

# Chapter 1  Basic Arithmetic and Functions

In the study of mathematics, a solid understanding of basic arithmetic operations and the concept of functions forms the foundation for more advanced topics. Arithmetic involves the manipulation of numbers through fundamental operations such as addition, subtraction, multiplication, and division. These operations are not only essential in everyday calculations but also serve as the building blocks for more complex mathematical procedures.

Functions, on the other hand, represent a crucial concept in mathematics, serving as a bridge between arithmetic and higher-level mathematical analysis. A function can be thought of as a special relationship between two sets, where each input (from the domain) is associated with exactly one output (from the co-domain). Understanding functions and their properties allows us to model and solve real-world problems with greater precision and flexibility.

In this section, we will explore the basic arithmetic rules and introduce the concept of functions, including their definitions, notations, and key properties. We will also discuss how these concepts are applied in various contexts, setting the stage for more advanced mathematical discussions.

## 1.1  Factorisation and the Order of Operations

Recall that when computing a product such as

$$a(b+c) = ab + ac$$

we distribute the factor $a$ across each term inside the parentheses. We call this **the distributive property**. However, it is often advantageous or necessary to perform the reverse operation, known as factorisation. Factorisation involves expressing the sum $ab + ac$ in its factored form $a(b+c)$.

Mathematically, the expression $a(b+c)$ is considered more simplified or "better" than $ab+ac$. To understand why, we must distinguish between **terms** and **factors**. Terms are separated by addition or subtraction, while factors are separated by multiplication or division. For instance, the expression

$$8 - 5 + 3$$

consists of three terms (8, -5, and 3), whereas the expression

$$2 \times 5 \times 3$$

consists of three factors (2, 5, and 3). Consider the expression

$$5 + 2 + 7 \times 8$$

which consists of three terms (5, 2, and $7 \times 8$), and note that one of the terms itself contains two factors (7 and 8). Similarly, the expression

$$5(2+4)(-9)$$

consists of three factors (5, $2 + 4$, and -9), with one factor, $2 + 4$, containing two terms.

The advantage of expressing mathematical expressions solely in terms of factors lies in the ability to simplify them more effectively. This concept will be elaborated throughout this chapter, especially when dealing with fractions and equations. Consider the following examples of factorisation:

$$8 - 5 + 3$$

consists of three terms (8, -5 and 3), while the expression

$$2 \times 5 \times 3$$

consists of three factors (2, 5 and 3). The expression

$$5 + 2 + 7 \times 8$$

consists of three terms (5, 2 and 7 $\times$ 8) and one of the terms consists of two factors (7 $\times$ 8), while the expression

$$5(2 + 4)(-9)$$

consists of three factors (5, (2 + 4), (-9)) and one of the factors consists of two terms ((2 + 4)).

It can be advantageous to express terms solely in factors because this allows us to simplify the expressions. This will become clearer as we progress through the lesson, and it is particularly important when working with fractions and equations. Here are some more examples of factorisation:

**Example 1.1** Factorisation

$$ab - ac = a(b - c)$$
$$-ab - ac = a(-b - c)$$
$$-ab - ac = -a(b + c)$$
$$ab - ac + aa - aa = a(b - c + a - a)$$
$$abc - ab + aba = ab(c - 1 + a)$$
$$ab - abc - a = a(b - bc - 1) = b(1 - c) - 1$$

When evaluating mathematical expressions, it is crucial to follow a specific order of operations to ensure accurate results. The correct sequence for performing these operations is as follows:

1. **Brackets (Parentheses)**: First, perform all operations inside brackets or parentheses.
2. **Exponents and Radicals**: Next, evaluate exponents (powers) and radicals (roots).
3. **Multiplication and Division**: Then, perform multiplication and division from left to right as they appear.
4. **Addition and Subtraction**: Finally, execute addition and subtraction from left to right as they appear.

Let's consider examples for each operation to illustrate the order of operations:

**Example 1.2** Brackets

Evaluate the expression: $(2 + 3) \times 4$

$$(2 + 3) \times 4 = 5 \times 4 = 20$$

**Example 1.3** Exponents and Radicals

Evaluate the expression: $3^2 + \sqrt{16}$

$$3^2 + \sqrt{16} = 9 + 4 = 13$$

**Example 1.4** Multiplication and Division

Evaluate the expression: $6 \div 2 \times 3$. According to the standard order of operations, we perform division and multiplication from left to right:

$$6 \div 2 \times 3 = (6 \div 2) \times 3 = 3 \times 3 = 9$$

**Example 1.5** Addition and Subtraction

Evaluate the expression: $8 - 3 + 2$

$$8 - 3 + 2 = 5 + 2 = 7$$

## 1.2 Fractions

By definition, a fraction always consists of (at least) two factors. The first factor we will call the **numerator** and is the "top part" of the fraction. The bottom part we will call the **denominator**. Perhaps the most important rule when working with fractions is that two fractions can only be added or subtracted if they have identical denominators. Also, the denominator must never be equal to 0.

**Example 1.6**

$$\frac{1}{x^2 - 2}$$

Here we must make sure that $x^2 - 2 \neq 0$ which means that we may only use values different from $\pm\sqrt{2}$.

> **Proposition 1.1 (Rules for Calculations with Fractions)**
>
> For $a, b, c, m \in \mathbb{R}$, with $a, b, c, m \neq 0$ where required, the following identities hold:
>
> $\quad$ (1) $\quad \dfrac{a}{b} \times m = \dfrac{am}{b}$
>
> $\quad$ (2) $\quad \dfrac{a}{b} \div m = \dfrac{a}{bm}$
>
> $\quad$ (3) $\quad m \div \dfrac{a}{b} = \dfrac{mb}{a}$
>
> $\quad$ (4) $\quad \dfrac{a}{b} \times \dfrac{c}{a} = \dfrac{c}{b}$
>
> $\quad$ (5) $\quad \dfrac{a}{b} \div \dfrac{c}{a} = \dfrac{a^2}{bc}$
>
> $\quad$ (6) $\quad \dfrac{a}{b} = \dfrac{ac}{bc}$
>
> $\quad$ (7) $\quad \dfrac{a}{b} + \dfrac{c}{a} = \dfrac{a^2 + bc}{ab}$

To extend or reduce a fraction, we must multiply or divide by the same numbers in the denominator and numerator:

**Example 1.7** Extending or reducing fractions

$$\frac{72}{144} = \frac{72 \div 12}{144 \div 12} = \frac{6}{12} = \frac{6 \div 6}{12 \div 6} = \frac{1}{2}$$

$$\frac{2}{3} = \frac{2 \times 6}{3 \times 6} = \frac{12}{18}$$

$$\frac{2x}{2xx} = \frac{2}{2x}$$

$$\frac{3a}{6a + 3b} = \frac{3a}{3(2a + b)} = \frac{a}{2a + b}$$

To factorise an expression, all terms must be divided or multiplied uniformly. This implies that it is not possible to simplify the following expression any further, even though it might be tempting:

$$\frac{a}{2a + b} \neq \frac{1}{2 + b}$$

For proper factorisation of $\frac{a}{2a+b}$, you must divide $a$ into all terms in the denominator:

$$\frac{a}{2a + b} = \frac{1}{2 + \frac{b}{a}}$$

As illustrated above, the multiplication of fractions is straightforward: you multiply the numerators and denominators with each other, respectively.

**Example 1.8** Multiplication of fractions

Consider the fractions $\frac{a}{b}$ and $\frac{c}{d}$. Their product is:

$$\frac{a}{b} \times \frac{c}{d} = \frac{a \times c}{b \times d}$$

For instance, if $a = 2$, $b = 3$, $c = 4$, and $d = 5$, then:

$$\frac{2}{3} \times \frac{4}{5} = \frac{2 \times 4}{3 \times 5} = \frac{8}{15}$$

**Example 1.9** Dividing fractions

$$\frac{6}{7} \div \frac{4}{21} = \frac{6}{7} \times \frac{21}{4} = \frac{126}{28} = \frac{63}{14} = \frac{9}{2}$$

$$\frac{1}{2} \div 3 = \frac{1}{2} \div \frac{3}{1} = \frac{1}{2} \times \frac{1}{3} = \frac{1}{6}$$

$$\frac{2}{3} \div \frac{8}{9} = \frac{2}{3} \times \frac{9}{8} = \frac{18}{24} = \frac{3}{4}$$

$$\frac{8}{9} \div 16 = \frac{8}{9} \times \frac{1}{16} = \frac{8}{144} = \frac{4}{72} = \frac{1}{18}$$

Adding and subtracting fractions seems to cause more problems than multiplication and division. The key is to find a common denominator between the fractions and then remember the above-mentioned rule about extending fractions.

**Example 1.10** Adding and subtracting fractions

$$\frac{1}{5} + \frac{2}{5} = \frac{1+2}{5} = \frac{3}{5}$$

$$\frac{1}{4} + \frac{2}{3} = \frac{1 \times 3}{4 \times 3} + \frac{2 \times 4}{3 \times 4} = \frac{3}{12} + \frac{8}{12} = \frac{3+8}{12} = \frac{11}{12}$$

$$\frac{7}{12} - \frac{5}{8} = \frac{7 \times 8}{12 \times 8} - \frac{5 \times 12}{8 \times 12} = \frac{56}{96} - \frac{60}{96} = \frac{56-60}{96} = \frac{-4}{96} = \frac{-1}{24}$$

Note: Never do

$$\frac{a}{b} + \frac{c}{d} = \frac{a+b}{b+d}$$

**Example 1.11**

$$\frac{x}{x-1} \times \frac{2}{x(x+4)} = \frac{2x}{(x-1)x(x+4)} = \frac{2}{(x-1)(x+4)}$$

$$\frac{2}{x-1} \div \frac{x}{x-1} = \frac{2}{x-1} \times \frac{x-1}{x} = \frac{2(x-1)}{(x-1)x} = \frac{2}{x}$$

$$\frac{x+1}{x^2+2} + \frac{x-6}{x^2+2} = \frac{x+1+x-6}{x^2+2} = \frac{2x-5}{x^2+2}$$

Remember, when a fraction is preceded by a minus sign, all signs in the numerator must be changed accordingly. This is similar to how you would change all signs within parentheses when they are preceded by a minus sign.

Consider the expression:

$$-\frac{a-b}{c}$$

To correctly handle the negative sign, change all the signs in the numerator:

$$-\frac{a-b}{c} = \frac{-a+b}{c}$$

For example, if $a = 5$ and $b = 3$, then:

$$-\frac{5-3}{c} = \frac{-5+3}{c} = \frac{-2}{c}$$

Example 1.12

$$\frac{x+1}{x^2+2} - \frac{x-6}{x^2+2} = \frac{x+1-x+6}{x^2+2} = \frac{7}{x^2+2}$$

## 1.3 Exponents, Radicals and Surds

An **exponent** is a shortcut for repeated multiplication of the same number:

**Example 1.13** Exponentiation

$$4 \times 4 \times 4 \times 4 \times 4 = 4^5$$

$$x \times x \times x \times x \times x = x^5$$

**Radicals**, or **roots**, represent the inverse operation of applying exponents. A radical is any number expressed with the radical symbol $\sqrt{\ }$. Specifically, applying a radical can reverse the effect of an exponent, and vice versa. For instance, squaring 2 yields 4, and taking the square root of 4 returns 2. Similarly, squaring 3 results in 9, and the square root of 9 brings us back to 3.

**Example 1.14** Taking the root

$$\sqrt{a} \times \sqrt{a} = (\sqrt{a})^2 = a$$

$$\sqrt{a} = b \implies (\sqrt{a})^2 = b^2 \iff a = b^2$$

A **surd** is a type of radical that is both real and irrational, examples include $\sqrt{2}, \sqrt{3}, \sqrt{5}$, and $\sqrt{6}$.

Numbers can be raised to powers other than 2, such as cubing (raising to the third power), or even raising to the fourth power, the 100th power, and so forth. Correspondingly, you can take the cube root of a number, the fourth root, the 100th root, and so on. To indicate a root other than a square root, the same radical symbol is used, but with a number called the index inserted into the radical sign, typically positioned within the "check mark" part.

**Example 1.15** Index and argument

$$4^3 = 64 \iff \sqrt[3]{64} = 4$$

In this example, the "3" inside the radical sign is the **index** of the radical. The "64" is referred to as the argument of the **radical**, also known as the **radicand**. Since square roots are the most common type of radicals, the index is usually omitted for square roots. Although "$\sqrt[2]{2}$" would be technically correct, it is rarely used in practice.

**Proposition 1.2 (Rules for calculations involving radicals)**

(1) $\sqrt[n]{xy} = \sqrt[n]{x} \times \sqrt[n]{y}$ where $x, y \geq 0$

(2) $\sqrt[n]{\dfrac{x}{y}} = \dfrac{\sqrt[n]{x}}{\sqrt[n]{y}}$ where $x \geq 0$ and $y > 0$

(3) $\sqrt{x^2} = |x|$ where $x \in \mathbb{R}$

(4) $\left(\sqrt[n]{x}\right)^n = x$ If $x < 0$ and $n \in \mathbb{N}$, then $\sqrt[n]{x}$ is not defined

(5) $\sqrt[n]{-x} = -\sqrt[n]{x}$ where $x \geq 0$ and $n \in \mathbb{N}$ is odd

Raising a number to a **power**, also known as **exponentiation**, is a fundamental mathematical operation that involves multiplying a number by itself a certain number of times as we saw above. The **base** is the number being multiplied, and the **exponent** indicates how many times the base is used as a factor. For example, $a^n$ means that the base $a$ is multiplied by itself $n$ times. Exponentiation is a powerful tool in mathematics, with a few essential rules that govern its application.

**Proposition 1.3 (Properties of Integer Exponents)**

Let $n, m \in \mathbb{Z}$. Then the following hold (with $x, y \in \mathbb{R}$ and nonzero where stated):

(1) $x^n \cdot x^m = x^{n+m}$,

(2) $\dfrac{x^n}{x^m} = x^{n-m}$ with $x \neq 0$,

(3) $x^n \cdot y^n = (xy)^n$,

(4) $\dfrac{x^n}{y^n} = \left(\dfrac{x}{y}\right)^n$ with $y \neq 0$,

(5) $\left(x^n\right)^m = x^{nm}$,

(6) $x^1 = x$.

Some of these rules allow the concept of powers to be extended so that the exponent may be any integer. If you set $n = m$ in rule (2), you get:

$$\frac{x^n}{x^n} = x^{n-n} = x^0$$

But since $\dfrac{x^n}{x^n} = 1$, we obtain

$$x^0 = 1$$

Thus, the concept of exponentiation is extended to include $n \in \mathbb{N} \cup \{0\}$. If you now set $n = 0$ again in rule (2), you get:

$$\frac{x^0}{x^m} = x^{0-m} = x^{-m}$$

But according to the previous calculation, $x^0 = 1$. Therefore, you obtain

$$\frac{1}{x^m} = x^{-m}$$

As demonstrated in Proposition A.3, these rules provide a complete framework for manipulating expressions with any integer exponents. As a consequence, the following additional rules can be derived:

> **Proposition 1.4 (More Properties of Integer Exponents)**
>
> Let $n, m \in \mathbb{Z}$. Then the following hold (with $x, y \in \mathbb{R}$ and nonzero where stated):
>
> (7)  $\quad x^0 = 1 \qquad\qquad x \neq 0$
>
> (8)  $\quad \dfrac{1}{x^m} = x^{-m} \qquad x \neq 0$

Let us illustrate these rules with a couple of examples.

**Example 1.16** Reduce the following expression

$$\left(3xy^6\right)^3 = 3^3 \cdot x^3 \cdot \left(y^6\right)^3 = 27x^3 y^{18}$$

**Example 1.17** Reduce the following expression

$$\frac{a^{-4}b^3}{a^7 b^{-5}} = \frac{a^{-4}}{a^7} \cdot \frac{b^3}{b^{-5}} = a^{-4-7} \cdot b^{3-(-5)} = a^{-11} \cdot b^8 = \frac{b^8}{a^{11}}$$

For positive numbers $x$, the concept of exponentiation can be further extended to apply when the exponent is a rational number. Any rational number $r \in \mathbb{Q}$ can be written as $r = \dfrac{m}{n}$, where $m \in \mathbb{Z}$ and $n \in \mathbb{N}$. For $x > 0$, we now define

$$y = x^r = x^{\frac{m}{n}}$$

From this, you obtain (using, among other things, rule (5)):

$$y^n = \left(x^{\frac{m}{n}}\right)^n = x^{\frac{m}{n} \cdot n} = x^m$$

Finally, by using the concept of radicals, we obtain

$$y^n = x^m \qquad \Longleftrightarrow \qquad y = \sqrt[n]{x^m}$$

Note that because $x > 0$, it follows that $y > 0$ as well. We are now ready to state **the extended concept of exponentiation**.

> **Definition 1.1 (The Extended Concept of Exponentiation)**
>
> Let $m \in \mathbb{Z}$ and let $n \in \mathbb{N}$ such that $\dfrac{m}{n} \in \mathbb{Q}$. Then the following applies:
>
> $x^{\frac{m}{n}} = \sqrt[n]{x^m} \qquad x > 0$
>
> And more specifically, the following holds true
>
> $x^{\frac{1}{n}} = \sqrt[n]{x} \qquad x > 0$

The denominator of a rational exponent corresponds to the index of the radical, while the numerator remains as the exponent of the base. Conversely, the index of a radical can be transformed into the denominator of an exponent in an equivalent exponential expression. This property allows us to convert any radical expression into an exponential form, providing a powerful tool for simplification.

**Example 1.18**

$$\sqrt[5]{x^3} = x^{\frac{3}{5}} \quad \text{vs.} \quad \sqrt[3]{x^5} = x^{\frac{5}{3}}$$

$$\frac{1}{\sqrt[7]{x^3}} = x^{-\frac{3}{7}}$$

$$\frac{1}{\sqrt[3]{x^2}} = \left(x^2\right)^{-\frac{2}{3}}$$

This property can also be reversed: any rational exponent can be rewritten as a radical expression by using the denominator as the radical's index. The ability to interchange between exponential and radical forms enables us to evaluate expressions that were previously difficult to handle by converting them into radicals.

### Example 1.19

$$27^{-\frac{4}{3}} = \frac{1}{\sqrt[3]{27^4}} = \frac{1}{\left(\sqrt[3]{27}\right)^4} = \frac{1}{3^4} = \frac{1}{81}$$

One of the greatest advantages of converting a radical expression into an exponential form is that it allows us to apply all the properties of exponents to simplify the expression. The following examples illustrate how various properties can be utilised to simplify expressions with rational exponents.

### Example 1.20

$$a^{\frac{2}{3}} b^{\frac{1}{2}} a^{\frac{1}{6}} b^{\frac{1}{5}} = a^{\frac{2}{3} + \frac{1}{6}} b^{\frac{1}{2} + \frac{1}{5}} = a^{\frac{4}{6} + \frac{1}{6}} b^{\frac{5}{10} + \frac{2}{10}} = a^{\frac{5}{6}} b^{\frac{7}{10}}$$

$$\left(x^{\frac{1}{3}} x^{\frac{2}{5}}\right)^{\frac{3}{4}} = x^{\frac{1}{3} \times \frac{3}{4}} x^{\frac{2}{5} \times \frac{3}{4}} = x^{\frac{3}{12}} x^{\frac{6}{20}} = x^{\frac{1}{4}} x^{\frac{3}{10}} = x^{\frac{11}{20}}$$

$$\frac{x^{\frac{4}{2}} x^{\frac{4}{6}} x^{\frac{1}{2}} x^{\frac{5}{6}}}{x^{\frac{7}{2}} x^0} = 2x^{\frac{4}{2} + \frac{1}{2}} x^{\frac{4}{6} + \frac{5}{6}} x^{\frac{7}{2}} = 2x^{\frac{5}{2}} x^{\frac{9}{6}} x^{\frac{7}{2}} = 2x^{-1} x^{\frac{3}{2}} = 2x^{\frac{1}{2}}$$

$$\left(25x^{\frac{1}{3}} x^{\frac{2}{5}}\right)^{-\frac{1}{2}} = \left(25x^{\frac{5}{15}} x^{\frac{4}{10}}\right)^{-\frac{1}{2}} = \left(25x^{-\frac{7}{15}} x^{\frac{19}{10}}\right)^{-\frac{1}{2}} = \left(\frac{9}{25x^{-\frac{7}{15}} x^{\frac{19}{10}}}\right)^{\frac{1}{2}} = \frac{9}{2} \cdot 25x^{-\frac{7}{30}} x^{\frac{19}{20}} = \frac{3x^{\frac{7}{30}}}{5x^{\frac{19}{20}}}$$

It is important to remember that when simplifying expressions with rational exponents, we are applying the same exponent rules that are used for integer exponents. The only difference is that we must also adhere to the rules for fractions.

## 1.4 Using Formulae and Substitution

In the study of engineering, physical quantities are often related to each other through formulas. These formulas consist of variables and constants that represent the physical quantities in question. To evaluate a formula, one must substitute numerical values for the variables.

For example, Ohm's law provides a formula that relates the voltage, $v$, across a resistor with a resistance value $R$, to the current $i$ flowing through it. The formula is given by

$$v = iR$$

This formula allows us to calculate the voltage $v$ if the values for $i$ and $R$ are known. For instance, if $i = 13\,\text{A}$ and $R = 5\,\Omega$, then

$$v = iR = (13)(5) = 65$$

Thus, the voltage is 65 V.

This example highlights the importance of paying close attention to the units of any physical quantities involved. A formula is only valid if a consistent set of units is used.

**Example 1.21** Inserting into formulae

The kinetic energy $K$ of an object with mass $M$ moving at speed $v$ can be calculated using the formula:

$$K = \frac{1}{2}Mv^2$$

Calculate the kinetic energy of an object with a mass of 5 kg moving at a speed of $2\,\mathrm{m\,s^{-1}}$.

*Solution:*

$$K = \frac{1}{2}Mv^2 = \frac{1}{2}(5)(2^2) = 10$$

◀

In the SI system, the unit of energy is the joule, so the kinetic energy of the object is 10 joules.

**Example 1.22** Inserting into formulae

The area $A$ of a circle with radius $r$ can be calculated using the formula $A = \pi r^2$.

Alternatively, if the diameter $d$ of the circle is known, the equivalent formula can be used:

$$A = \frac{\pi d^2}{4}$$

Calculate the area of a circle with a diameter of 0.1 m. The value of $\pi$ is pre-programmed in your calculator.

*Solution:*

$$A = \frac{\pi(0.1)^2}{4} = 0.00785\,\mathrm{m^2}$$

◀

**Example 1.23** Inserting into formulae

The volume $V$ of a circular cylinder is equal to its cross-sectional area $A$ multiplied by its length $h$.

Calculate the volume of a cylinder with a diameter of 0.1 m and a length of 0.3 m.

*Solution:*

$$V = Ah = \frac{\pi(0.1)^2}{4} \times 0.3 = 0.00236$$

The volume is $0.00236\,\mathrm{m^3}$.

◀

## 1.5 Rearranging Formulae

In the formula for the area of a circle, $A = \pi r^2$, the variable $A$ is referred to as the subject of the formula. A variable is considered the subject if it appears by itself on one side of the equation, usually on the left-hand side, and nowhere else in the formula. If we are asked to transpose the formula for $r$, or solve for $r$, we must rearrange the equation so that $r$ becomes the subject. When transposing a formula, any operation

performed on one side must also be applied to the other side. There are five key rules to follow during this process.

> **Rules for rearranging formulae**
>
> The following operations can be performed on both sides of the formula:
> - Add the same quantity to both sides
> - Subtract the same quantity from both sides
> - Multiply both sides by the same quantity - remember to multiply all terms
> - Divide both sides by the same quantity - remember to divide all terms
> - Apply a function to both sides, such as squaring or finding the reciprocal

**Example 1.24** Transpose the formula $p = 5t - 17$ to make $t$ the subject.

*Solution:* To isolate $t$ on the left-hand side, proceed in steps using the five rules. First, add 17 to both sides of the equation $p = 5t - 17$:

$$p + 17 = 5t - 17 + 17$$

Simplifying, we get:

$$p + 17 = 5t$$

Next, divide both sides by 5 to isolate $t$:

$$\frac{p + 17}{5} = t$$

Thus, the formula for $t$ is:

$$t = \frac{p + 17}{5}$$

◀

**Example 1.25** Transpose the formula $\sqrt{2q} = p$ to solve for $q$.

*Solution:* First, square both sides to eliminate the square root around $2q$. Note that $(\sqrt{2q})^2 = 2q$. This gives:

$$2q = p^2$$

Next, divide both sides by 2 to solve for $q$:

$$q = \frac{p^2}{2}$$

◀

**Problem 1.1** Transpose the formula $v = \sqrt{t^2 + w}$ to solve for $w$. To isolate $w$, follow these steps:

a. First, square both sides to eliminate the square root around $t^2 + w$:
$$v^2 = t^2 + w$$

b. Next, subtract $t^2$ from both sides to isolate $w$:
$$v^2 - t^2 = w$$

c. Finally, write down the formula for $w$:
$$w = v^2 - t^2$$

**Example 1.26** Transpose the formula $x = \frac{1}{y}$ to solve for $y$.

*Solution:* To isolate $y$, notice that $y$ appears in the denominator. Multiplying both sides by $y$ removes the fraction:
$$yx = y \times \frac{1}{y}$$
This simplifies to:
$$yx = 1$$
Finally, divide both sides by $x$ to solve for $y$:
$$y = \frac{1}{x}$$
Alternatively, you can simply invert both sides directly to obtain:
$$y = \frac{1}{x}$$
◀

**Example 1.27** Make $R$ the subject of the formula:
$$\frac{2}{R} = \frac{3}{x+y}$$

*Solution:* Since $R$ appears in a fraction, invert both sides:
$$\frac{R}{2} = \frac{x+y}{3}$$
Multiplying both sides by 2 yields: $R = \dfrac{2(x+y)}{3}$   ◀

**Example 1.28** Make $R$ the subject of the formula:
$$\frac{1}{R} = \frac{1}{R_1} + \frac{1}{R_2}$$

*Solution:* The two terms on the right-hand side can be combined:
$$\frac{1}{R_1} + \frac{1}{R_2} = \frac{R_2 + R_1}{R_1 R_2}$$
The formula then becomes:
$$\frac{1}{R} = \frac{R_2 + R_1}{R_1 R_2}$$
Finally, inverting both sides gives:
$$R = \frac{R_1 R_2}{R_2 + R_1}$$
◀

## 1.6 Functions

In mathematics, a function assigns each element of one set to a specific element of another set (which may be the same set). For example, consider a Mathematics for Software Engineering class where each student is assigned a grade from the set $\{12, 10, 7, 4, 02\}$. Suppose the grades are as follows:
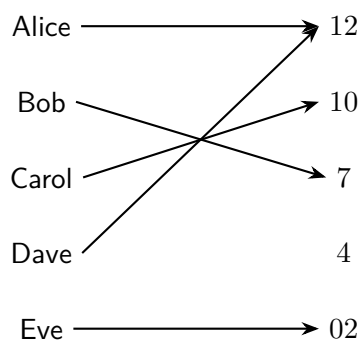


Figure 1.1: Example of a function mapping names to numbers.

This assignment of grades, illustrated in Figure 1.1, exemplifies a function.

Functions play a crucial role in mathematics and computer science. They define discrete structures such as sequences and strings and are used to analyse the time complexity of algorithms. Many computer programs are designed to compute values of functions. Recursive functions, defined in terms of themselves, are especially significant in computer science. This section provides an overview of the fundamental concepts of functions needed in the mathematics for software engineering.

> **Definition 1.2**
>
> Let $A$ and $B$ be nonempty sets. A function $f$ from $A$ to $B$ is an assignment of exactly one element of $B$ to each element of $A$. We write $f(a) = b$ if $b$ is the unique element of $B$ assigned by the function $f$ to the element $a$ of $A$. If $f$ is a function from $A$ to $B$, we write $f : A \to B$. ♣

**Remark:** Functions are sometimes also called mappings or transformations.

Functions can be specified in various ways. Sometimes, we explicitly state the assignments, as shown in Figure 1.1. Often, a formula such as $f(x) = x + 1$ is used to define a function. In other cases, a computer program may specify the function.

> **Definition 1.3**
>
> If $f$ is a function from $A$ to $B$, we say that $A$ is the **domain** of $f$ and $B$ is the **co-domain** of $f$. If $f(a) = b$, we say that $b$ is the **image** of $a$ and $a$ is a **preimage** of $b$. The **range**, or image, of $f$ is the set of all images of elements of $A$. Also, if $f$ is a function from $A$ to $B$, we say that $f$ **maps** $A$ to $B$. ♣

When defining a function, we specify its domain, co-domain, and the mapping of elements from the domain to the co-domain. Two functions are equal if they have the same domain, the same co-domain, and map each element of their domain to the same element in the co-domain.

It's important to note that altering the domain or co-domain results in a different function. Similarly, changing the mapping of elements also produces a different function.
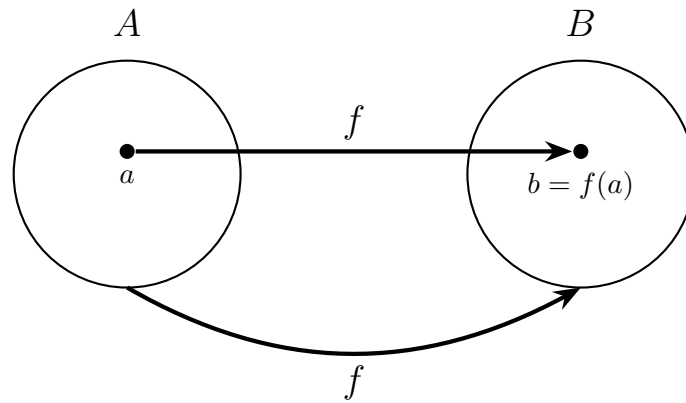
**Figure 1.2:** A function $f$ mapping an element $a$ from set $A$ to an element $b = f(a)$ in set $B$.

The following examples illustrate various functions. In each example, we describe the domain, co-domain, range, and the assignment of values to the elements of the domain.

**Example 1.29** What are the domain, co-domain, and range of the function that assigns grades to students described in the first paragraph of the introduction of this section?

*Solution:* Let $G$ be the function that assigns a grade to a student in our Software engineering mathematics class. Note that $G(\text{Alice}) = 12$, for instance. The domain of $G$ is the set {Alice, Bob, Carol, David, Eve }, and the co-domain is the set $\{12, 10, 7, 4, 02\}$. The range of $G$ is the set $\{12, 10, 7, 02\}$, because each grade except $4$ is assigned to some student. ◄

**Example 1.30** Let $f$ be the function that assigns the last two bits of a bit string of length 2 or greater to that string. For example, $f(11010) = 10$. Then, the domain of $f$ is the set of all bit strings of length 2 or greater, and both the co-domain and range are the set $\{00, 01, 10, 11\}$.

**Example 1.31** Let $f : \mathbb{Z} \to \mathbb{Z}$ assign the square of an integer to this integer. Then, $f(x) = x^2$, where the domain of $f$ is the set of all integers, the co-domain of $f$ is the set of all integers, and the range of $f$ is the set of all integers that are perfect squares, namely, $\{0, 1, 4, 9, \ldots\}$.

## One-to-One and Onto Functions

In mathematics, functions are a fundamental concept used to describe the relationship between two sets. However, not all functions behave the same way. To understand these differences, we introduce the concepts of one-to-one (injective) and onto (surjective) functions.

Some functions never assign the same value to two different domain elements. These functions are said to be **one-to-one**.

> **Definition 1.4 (One-to-One functions (Injective))**
>
> A function $f : A \to B$ is called **one-to-one** (or **injective**) if different elements in $A$ map to different elements in $B$. In other words, if $f(a_1) = f(a_2)$, then $a_1 = a_2$. This property ensures that no two distinct elements in $A$ are mapped to the same element in $B$. ♣

Graphically, a function is one-to-one if no horizontal line intersects the graph of the function at more than one point.

> **Definition 1.5 (Onto Functions (Surjective))**
>
> A function $f : A \to B$ is called **onto** (or **surjective**) if every element in $B$ is the image of at least one element in $A$. In other words, for every $b \in B$, there exists at least one $a \in A$ such that $f(a) = b$. This property ensures that the function "covers" the entire set $B$. ♣

## Inverse Functions

Now, consider a function $f : A \to B$ that is both one-to-one and onto. Because $f$ is onto, every element of $B$ is the image of some element in $A$. Furthermore, because $f$ is one-to-one, every element of $B$ is the image of a unique element of $A$. This unique correspondence allows us to define a new function from $B$ to $A$ that "reverses" the mapping given by $f$.
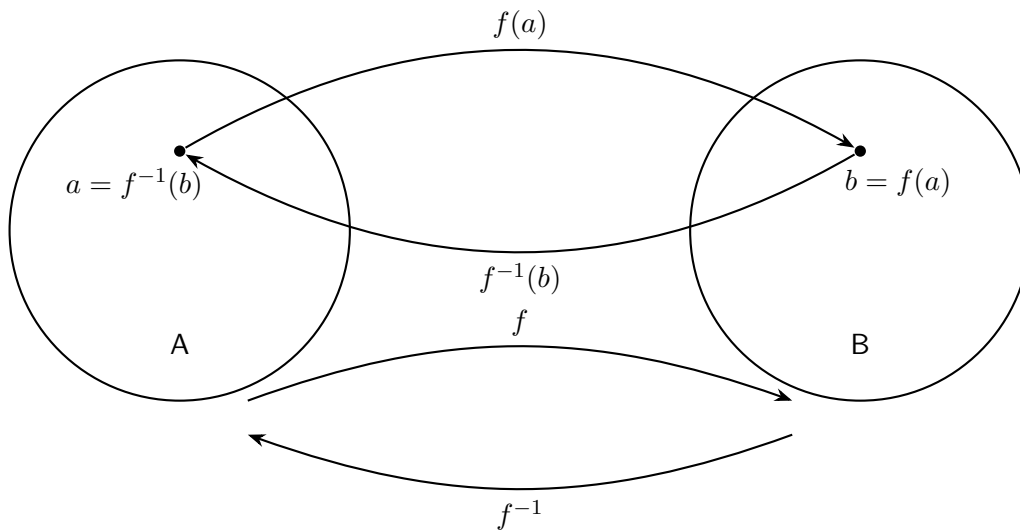


**Figure 1.3:** The function $f^{-1}$ is the inverse of function $f$.

This new function is called the **inverse function** of $f$, denoted by $f^{-1} : B \to A$. The inverse function $f^{-1}$ satisfies the following properties:

$$f(f^{-1}(b)) = b \quad \text{for every } b \in B.$$

$$f^{-1}(f(a)) = a \quad \text{for every } a \in A.$$

We can summarise these considerations in the following definition

> **Definition 1.6 (Inverse Functions)**
>
> Let $f$ be a one-to-one correspondence from the set $A$ to the set $B$. The inverse function of $f$ is the function that assigns to an element $b$ belonging to $B$ the unique element $a$ in $A$ such that $f(a) = b$. The inverse function of $f$ is denoted by $f^{-1}$. Hence, $f^{-1}(b) = a$ when $f(a) = b$. ♣

These properties show that $f^{-1}$ effectively undoes the work of $f$, mapping each element of $B$ back to the corresponding element in $A$.

**Example 1.32**

Let $f : \mathbb{R} \to \mathbb{R}$ be defined by $f(x) = 2x + 3$. We can check that $f$ is both one-to-one and onto:

- **One-to-One**: If $f(x_1) = f(x_2)$, then $2x_1 + 3 = 2x_2 + 3$. Subtracting 3 from both sides gives $2x_1 = 2x_2$, and dividing by 2 yields $x_1 = x_2$. Thus, $f$ is one-to-one.
- **Onto**: Given any $y \in \mathbb{R}$, we can solve $y = 2x + 3$ for $x$ to find $x = \frac{y-3}{2}$. Since this $x$ exists for every $y$, $f$ is onto.

Since $f$ is both one-to-one and onto, it has an inverse function $f^{-1}$ defined by

$$f^{-1}(y) = \frac{y - 3}{2}.$$

**Composite Functions**

In mathematics, functions can be combined to form new functions. One important way of combining functions is through the composition of functions. The composite of two functions is essentially applying one function to the results of another.
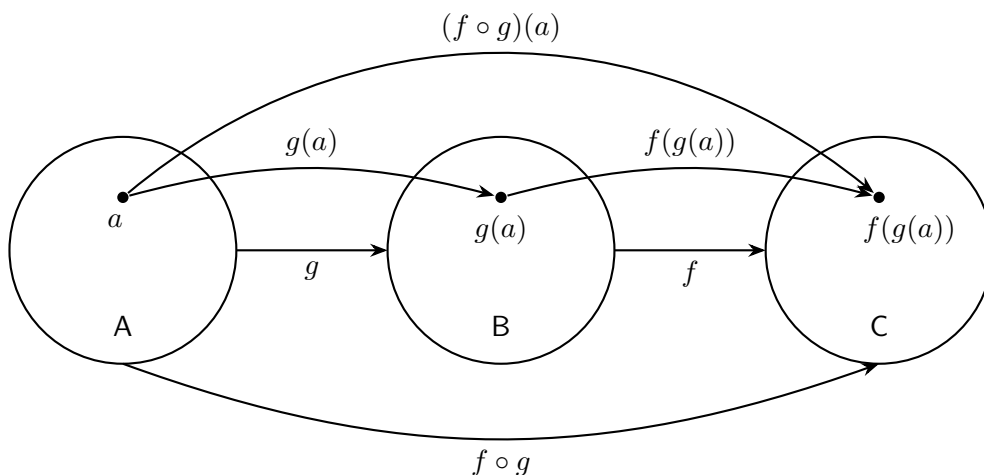


**Figure 1.4:** The composition of $f$ and $g$.

---
**Definition 1.7**

Let $f : B \to C$ and $g : A \to B$ be two functions. The **composite function** of $f$ and $g$, denoted by $f \circ g$, is a function from $A$ to $C$ defined by

$$(f \circ g)(x) = f(g(x)),$$

for every $x \in A$.

♣

---

In other words, the composite function $f \circ g$ means that you first apply the function $g$ to the input $x$, and then apply the function $f$ to the result of $g(x)$. In Figure 1.4 the composition of functions is shown.

**Example 1.33**

Consider the functions $f(x) = 2x + 3$ and $g(x) = x^2$. The composite function $f \circ g$ is given by:

$$(f \circ g)(x) = f(g(x)) = f(x^2) = 2x^2 + 3.$$

Here, the function $g(x)$ squares the input $x$, and then the function $f(x)$ multiplies the result by 2 and adds 3.

Now, let's reverse the composition and compute $g \circ f$:

$$(g \circ f)(x) = g(f(x)) = g(2x + 3) = (2x + 3)^2.$$

Notice that $f \circ g$ and $g \circ f$ are generally different functions, illustrating that the composition of functions is not commutative.

**Example 1.34** Let $g$ be the function from the set $\{a, b, c\}$ to itself such that $g(a) = b$, $g(b) = c$, and $g(c) = a$. Let $f$ be the function from the set $\{a, b, c\}$ to the set $\{1, 2, 3\}$ such that $f(a) = 3$, $f(b) = 2$, and $f(c) = 1$. What is the composition of $f$ and $g$, and what is the composition of $g$ and $f$ ?

*Solution:* The composition $f \circ g$ is defined by $(f \circ g)(a) = f(g(a)) = f(b) = 2$, $(f \circ g)(b) = f(g(b)) = f(c) = 1$, and $(f \circ g)(c) = f(g(c)) = f(a) = 3$.

Note that $g \circ f$ is not defined, because the range of $f$ is not a subset of the domain of $g$. ◀

**Example 1.35** label: Let $f$ and $g$ be the functions from the set of integers to the set of integers defined by $f(x) = 2x + 3$ and $g(x) = 3x + 2$. What is the composition of $f$ and $g$ ? What is the composition of $g$ and $f$?

*Solution:* Both the compositions $f \circ g$ and $g \circ f$ are defined. Moreover,

$$(f \circ g)(x) = f(g(x)) = f(3x + 2) = 2(3x + 2) + 3 = 6x + 7$$

and

$$(g \circ f)(x) = g(f(x)) = g(2x + 3) = 3(2x + 3) + 2 = 6x + 11.$$

◀

**Remark:** Even though $f \circ g$ and $g \circ f$ are defined for the functions $f$ and $g$ in example 35, $f \circ g$ and $g \circ f$ are not equal. In other words, the commutative law does not hold for the composition of functions.

When the composition of a function and its inverse is formed, in either order, an identity function is obtained. To see this, suppose that $f$ is a one-to-one correspondence from the set $A$ to the set $B$. Then the inverse function $f^{-1}$ exists and is a one-to-one correspondence from $B$ to $A$. The inverse function reverses the correspondence of the original function, so $f^{-1}(b) = a$ when $f(a) = b$, and $f(a) = b$ when $f^{-1}(b) = a$.

Hence,

$$\left(f^{-1} \circ f\right)(a) = f^{-1}(f(a)) = f^{-1}(b) = a,$$

and

$$\left(f \circ f^{-1}\right)(b) = f\left(f^{-1}(b)\right) = f(a) = b.$$

Consequently, $f^{-1} \circ f = I_A$ and $f \circ f^{-1} = I_B$, where $I_A$ and $I_B$ are the identity functions on the sets $A$ and $B$, respectively. That is, $\left(f^{-1}\right)^{-1} = f$.

**Example 1.36** If $f : \mathbb{R} \to \mathbb{R}$ is defined as $f(x) = 2x + 3$, then $f^{-1}(x) = \frac{x-3}{2}$. The composition $f \circ f^{-1}$ would be the identity function $I_\mathbb{R}$ on the real numbers, meaning $f\left(f^{-1}(x)\right) = x$ for all $x \in \mathbb{R}$.

## 1.7 Graphical Identification of Function Types

Understanding the behaviour of different types of functions is fundamental in mathematics. Functions can be classified based on their graphical patterns, which provide valuable insights into their characteristics. In this section, we will explore various types of functions, including linear, quadratic, exponential, and more. By examining their graphs, we can identify key features such as intercepts, slopes, curvature, and asymptotic behaviour, enabling us to distinguish between these different types of functions effectively.

### Linear Functions

The general equation for a linear function is given by

$$y = ax + b \quad \text{(often written as } y = mx + b\text{)},$$

where $a$ (or $m$) represents the slope and $b$ is the y-intercept. The domain of this function is all real numbers. This equation is in slope-intercept form because $a$ (or $m$) gives the slope and $b$ gives the y-intercept. If $a = 0$, the function simplifies to $y = b$, which is a constant function.

The parent function for a linear equation is

$$y = x.$$

The transformed function can be written in the point-slope form as

$$y = y_1 + a(x - x_1),$$

where the graph contains the point $(x_1, y_1)$ and has slope $a$. In this form:

- $a$ is the vertical dilation (slope),
- $y_1$ represents the vertical translation,
- $x_1$ represents the horizontal translation.

This point-slope form can also be written as

$$y - y_1 = a(x - x_1),$$

where the coordinates of the fixed point $(x_1, y_1)$ appear with a negative sign. The form $y = y_1 + a(x - x_1)$ expresses $y$ explicitly in terms of $x$, making it easier to enter into a graphing calculator.
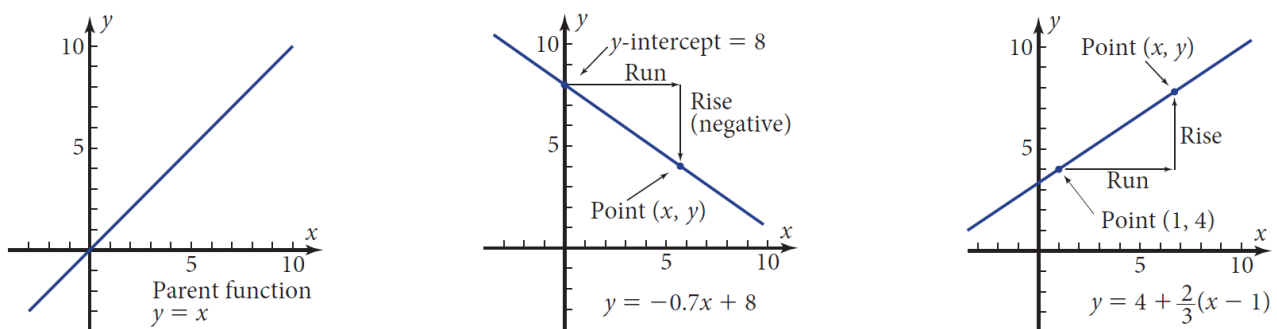


Figure 1.5: Linear functions

The graph of a linear function is a straight line. The parent function $y = x$ is shown on the left in Figure 1.5, the slope-intercept form in the middle, and the point-slope form on the right.

For the slope-intercept form: "Start at $b$ on the $y$-axis, move $x$ units horizontally, and rise $ax$ units vertically." For the point-slope form: "Start at $(x_1, y_1)$, move $(x - x_1)$ units horizontally, and rise $a(x - x_1)$ units vertically."

## Quadratic Functions

The general equation for a quadratic function is given by

$$y = ax^2 + bx + c,$$

where $a \neq 0$, and $a$, $b$, and $c$ are constants. The domain of this function is all real numbers.
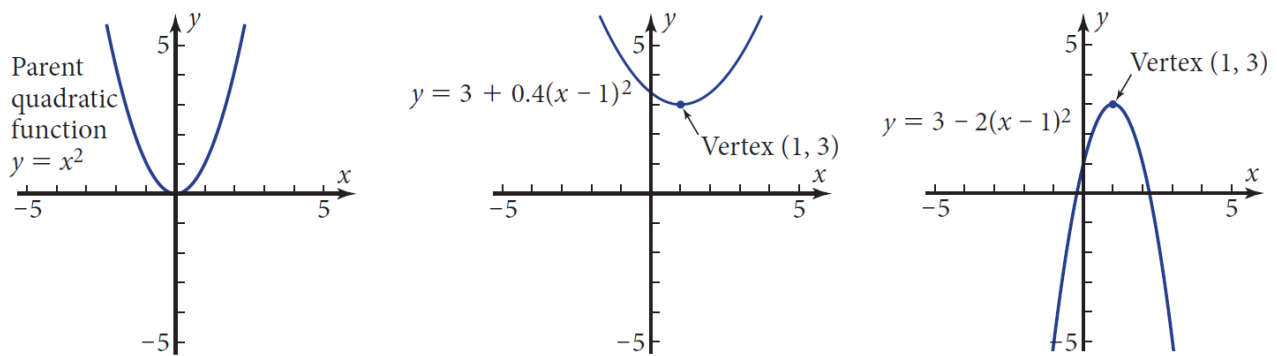


**Figure 1.6:** Quadratic functions

The parent function for a quadratic equation is

$$y = x^2,$$

where the vertex of the parabola is at the origin $(0, 0)$.

The transformed function can be written in vertex form as

$$y = k + a(x - h)^2,$$

where the vertex of the parabola is located at $(h, k)$. In this form:

- $k$ represents the vertical translation,
- $h$ represents the horizontal translation,
- $a$ represents the vertical dilation.

Vertex form can also be written as

$$y - k = a(x - h)^2,$$

but expressing $y$ explicitly in terms of $x$ makes the equation easier to enter into a graphing calculator.

The graph of a quadratic function is a parabola (from the Greek word for "along the path of a ball"). The parabola is concave up if $a > 0$ and concave down if $a < 0$. This behaviour is illustrated in Figure 1.6.

## Power Functions

The general equation for a power function is given by

$$y = ax^b,$$

where $a$ and $b$ are nonzero constants. The domain of the function depends on the value of $b$:

- If $b > 0$, the domain is all real numbers.
- If $b < 0$, the domain excludes $x = 0$ to avoid division by zero.
- If $b$ is not an integer, the domain usually excludes negative numbers to avoid taking roots of negative numbers.

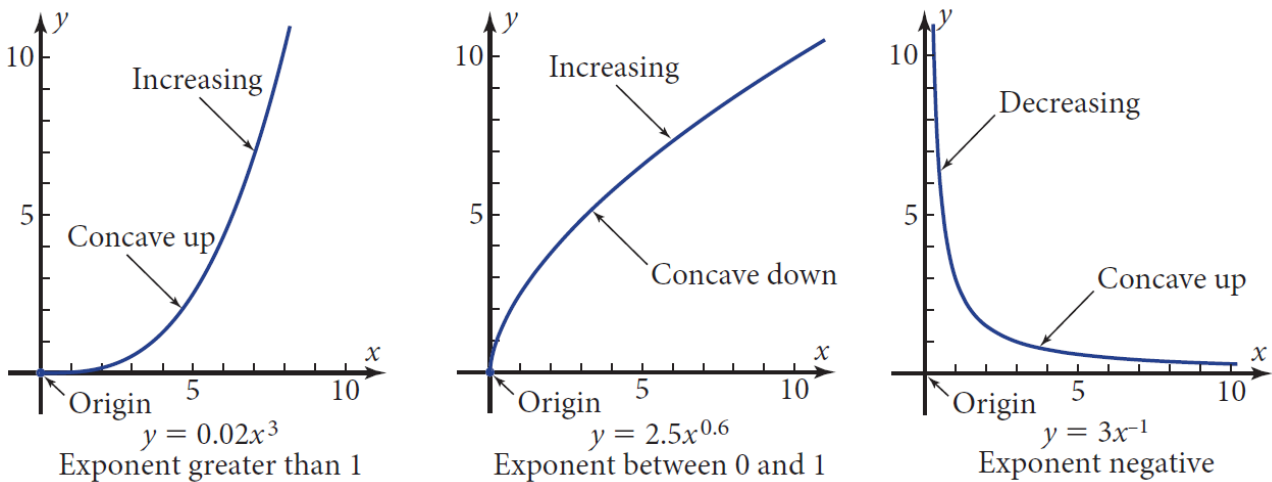In most applications, the domain is restricted to non-negative numbers.



**Figure 1.7:** Power functions

The parent function for a power function is

$$y = x^b.$$

For the general power function $y = ax^b$:

- If $b > 0$, then $y$ varies directly with the $b$th power of $x$, meaning $y$ is directly proportional to the $b$th power of $x$.
- If $b < 0$, then $y$ varies inversely with the $b$th power of $x$, meaning $y$ is inversely proportional to the $b$th power of $x$.

The dilation factor $a$ serves as the proportionality constant.

The translated form of a power function is

$$y = d + a(x - c)^b,$$

where $c$ and $d$ are the horizontal and vertical translations, respectively. This can be compared with the translated forms of linear and quadratic functions:

$$y = y_1 + a(x - x_1) \qquad \text{(linear function)},$$
$$y = k + a(x - h)^2 \quad \text{(quadratic function)}.$$

Unless otherwise stated, "power function" will imply the untranslated form, $y = ax^b$.

Figure A.1 shows the graphs of power functions for different values of $b$. In all cases, $a > 0$. The shape and concavity of the graph depend on the value of $b$:

- If $b > 0$, the graph contains the origin.

- If $b < 0$, the graph has the axes as asymptotes.
- The function is increasing if $b > 0$ and decreasing if $b < 0$.
- The graph is concave up if $b > 1$ or $b < 0$, and concave down if $0 < b < 1$.

The concavity of the graph describes the rate at which $y$ increases. For $b > 0$, concave up indicates that $y$ is increasing at an increasing rate, while concave down indicates that $y$ is increasing at a decreasing rate.

## Exponential Functions

The general equation for an exponential function is given by

$$y = ab^x,$$

where $a$ and $b$ are constants, $a \neq 0$, $b > 0$, and $b \neq 1$. The domain of this function is all real numbers.

The parent function for an exponential equation is

$$y = b^x,$$
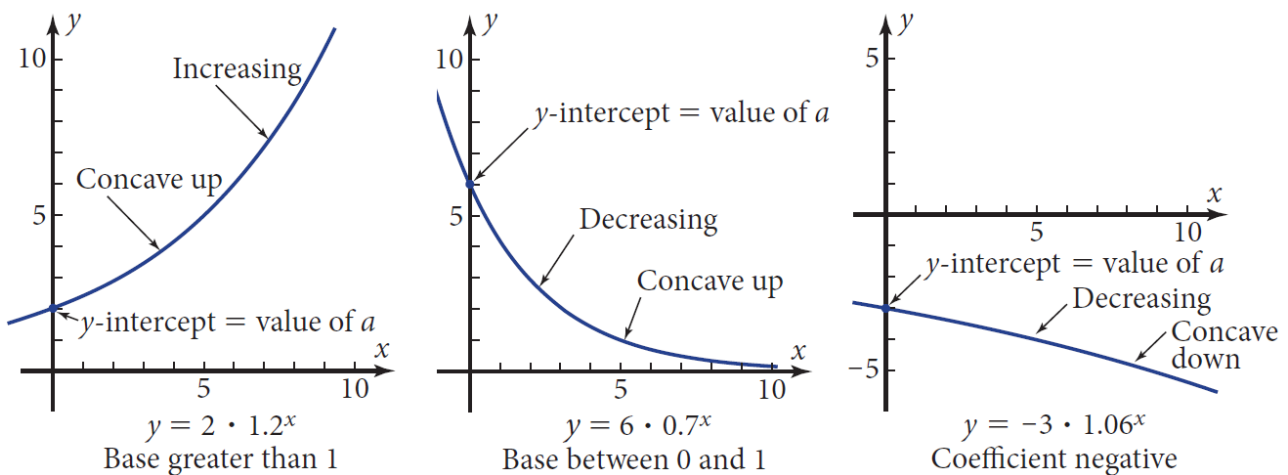
where the asymptote is the $x$-axis.



**Figure 1.8:** Exponential functions

In the equation $y = ab^x$, we say that "$y$ varies exponentially with $x$." This means that $y$ changes by a constant factor $b$ for each unit increase in $x$.

The translated form of the exponential function is

$$y = ab^x + c,$$

where the asymptote is the line $y = c$. Unless otherwise stated, "exponential function" will refer to the untranslated form $y = ab^x$.

Figure A.2 illustrates exponential functions for different values of $a$ and $b$. The key properties of the graph are as follows:

- The constant $a$ is the $y$-intercept of the graph.
- The function is increasing if $b > 1$ and decreasing if $0 < b < 1$, provided $a > 0$.
- If $a < 0$, the function's behavior is reversed: it is decreasing if $b > 1$ and increasing if $0 < b < 1$.

- The graph is concave up if $a > 0$ and concave down if $a < 0$.

Mathematicians often use one of two particular constants as the base for an exponential function: either 10, which is the base of the decimal system, or the naturally occurring number $e$, which approximately equals 2.71828. These bases are significant in various mathematical applications.

> **Definition 1.8 (Special Exponential Functions)**
>
> $\quad y = a \cdot 10^{bx} \qquad$ base-10 exponential function
>
> $\quad y = a \cdot e^{bx} \qquad$ natural (base-$e$) exponential function,
>
> where $a$ and $b$ are constants and the domain is all real numbers. ♣

To generalise the exponential function, the variable in the exponent is often multiplied by a constant. The (untranslated) general forms of these exponential functions are given below:

$$y = a \cdot 10^{bx} \quad \text{and} \quad y = a \cdot e^{bx}$$

These functions can be further generalised by incorporating translations in both the $x$- and $y$-directions. The translated forms are:

$$y = a \cdot 10^{b(x-c)} + d \quad \text{and} \quad y = a \cdot e^{b(x-c)} + d$$

The base-$e$ exponential function, in particular, has a significant advantage when studying calculus, as the rate of change of $e^x$ is equal to $e^x$ itself.

## 1.8 Logarithms

Any positive number can be written as a power of 10. For instance,

$$3 = 10^{0.477\dots}$$
$$5 = 10^{0.6989\dots}$$
$$15 = 10^{1.1760\dots}$$

The exponents $0.4771\dots$, $0.6989\dots$, and $1.1760\dots$ are called the base-10 logarithms of 3, 5, and 15, respectively:

$$\log 3 = 0.4771\dots$$
$$\log 5 = 0.6989\dots$$
$$\log 15 = 1.1760\dots$$

To better understand the meaning of logarithms, press `LOG 3` on your calculator. You will get:

$$\log 3 = 0.477121254\dots$$

Then, without rounding, raise 10 to this power. You will obtain:

$$10^{0.477121254\dots} = 3$$

The powers of 10 have the normal properties of exponentiation. For instance,

$$15 = (3)(5) = \left(10^{0.4771...}\right)\left(10^{0.6999...}\right)$$
$$= 10^{0.4771...+0.6599...}$$
$$= 10^{1.1760...}$$

This means $10^{0.4771...+0.6599...} = 10^{1.1760...}$. Here, you add the exponents while keeping the same base. You can verify with your calculator that $10^{1.1760...}$ indeed equals 15.

From this example, you can infer that logarithms have the same properties as exponents. This is expected because logarithms *are* exponents. For instance,

$$\log(3 \cdot 5) = \log 3 + \log 5 \quad \textit{The logarithm of a product equals the sum of the logarithms of the factors.}$$

From the values given earlier, you can also show that:

$$\log \frac{15}{3} = \log 15 - \log 3 \quad \textit{The logarithm of a quotient.}$$

This property is reasonable because you divide powers of equal bases by subtracting the exponents:

$$\frac{15}{3} = \frac{10^{1.1760...}}{10^{0.477...}} = 10^{1.1760...-0.4771...} = 10^{0.6989...} = 5$$

Since a power can be written as a product, you can find the logarithm of a power as follows:

$$\log 34 = \log(3 \cdot 3 \cdot 3 \cdot 3) = \log 3 + \log 3 + \log 3 + \log 3$$
$$= 4 \log 3 \quad \textit{Combine like terms.}$$

The logarithm of a power equals the exponent of that power times the logarithm of the base. To verify this result, observe that $3^4 = 81$. Press $4 \times \text{LOG}\, 3$ on your calculator, and you'll find it equals $1.9084\ldots$.

> **Definition 1.9 (Base-10 Logarithms)**
>
> $$\log x = y \iff 10^y = x$$
>
> *Verbally*: $\log x$ is the exponent in the power of 10 that gives $x$ ♣

The term logarithm comes from the Greek words *logos*, meaning "ratio," and *arithmos*, meaning "number." Before the invention of calculators, base-10 logarithms were calculated approximately using infinite series and recorded in tables. Products involving many factors, such as

$$(357)(4.367)(22.4)(3.142)$$

could be calculated by adding their logarithms (exponents) rather than tediously multiplying several pairs of numbers. This method was invented by Englishman Henry Briggs (1561–1630) and Scotsman John Napier (1550–1616). The name logarithm, thus, reflects this "logical way to do arithmetic".

> **Properties of base-10 logarithms**
>
> - Log of a Product:
>   $$\log xy = \log x + \log y$$
>   *Verbally*: The $\log$ of a product equals the sum of the logs of the factors.
>
> - Log of a Quotient:
>   $$\log \frac{x}{y} = \log x - \log y$$
>   *Verbally*: The $\log$ of a quotient equals the log of the numerator minus the $\log$ of the denominator.
>
> - Log of a Power:
>   $$\log x^y = y \log x$$
>   *Verbally*: The $\log$ of a power equals the exponent times the log of the base.

**Example 1.37** Find $x$ if $\log_{10} 10^{3.721} = x$

*Solution:* By definition, the logarithm is the exponent of 10. So $x = 3.721$. ◀

**Example 1.38** Find $x$ if $0.258 = 10^x$

*Solution:* By definition, $x$, the exponent of 10 , is the logarithm of 0.258 .

$$x = \log_{10} 0.258 = -0.5883\ldots$$

◀

> **The most important thing to remember about logarithms is this**
>
> **A logarithm is an exponent.**

### Logarithms with Any Base: The Change-of-Base Property

If $x = 10^y$, then $y$ is the base-10 logarithm of $x$. Similarly, if $x = 2^y$, then $y$ is the base-2 logarithm of $x$. The only difference between these logarithms is the number that serves as the base. To distinguish among logarithms with different bases, the base is written as a subscript after the abbreviation "log." For instance:

$$3 = \log_2 8 \Leftrightarrow 2^3 = 8,$$
$$4 = \log_3 81 \Leftrightarrow 3^4 = 81,$$
$$2 = \log_{10} 100 \Leftrightarrow 10^2 = 100.$$

The symbol $\log_2 8$ is pronounced "log to the base 2 of 8." The symbol $\log_{10} 100$ is, of course, equivalent to $\log 100$, as defined in the previous section. Note that in all cases, a logarithm represents an exponent.

> **Definition 1.10 (Logarithm with Any Base)**
>
> *Algebraically*:
> $$\log_b x = y \text{ if and only if } b^y = x, \quad \text{where } b > 0, b \neq 1, \text{ and } x > 0$$
>
> *Verbally*:

> $\log_b x = y$ means that $y$ is the exponent of $b$ that gives $x$ as the answer. ♣

The way you pronounce the symbol for logarithm gives you a way to remember the definition. The next two examples show you how to do this.

**Example 1.39** Write $\log_5 c = a$ in exponential form.

*Solution:*

Think this:

- "$\log_5 \ldots$" is read as "log base 5 $\ldots$," meaning 5 is the base.
- A logarithm is an exponent. Since the $\log$ equals $a$, $a$ must be the exponent.
- The "answer" obtained from $5^a$ is the argument of the logarithm, denoted as $c$.

Write only this:

$$5^a = c$$

◀

**Example 1.40** Write $z^4 = m$ in logarithmic form.

*Solution:*  $\log_z m = 4$ ◀

Two bases of logarithms are used frequently enough to have their own key on most calculators. One is the base-10 logarithm, also known as the common logarithm, as discussed in the previous section. The other is the base-$e$ logarithm, known as the natural logarithm, where $e = 2.71828\ldots$, a naturally occurring number (like $\pi$) that will be advantageous in your future mathematical studies.

The symbol $\ln x$ (pronounced "el en of $x$") is used for natural logarithms, and is defined as:

$$\ln x = \log_e x$$

> **Definition 1.11 (Common Logarithm and Natural Logarithm)**
> *Common*: The symbol $\log x$ means $\log_{10} x$.
> *Natural*:   The symbol $\ln x$ means $\log_e x$, where $e$ is a constant equal to $2.71828182845\ldots$ ♣

**Example 1.41** Find $\log_5 17$. Check your answer by an appropriate numerical method.

*Solution:*  Let $x = \log_5 17$.

$$5^x = 17$$
$$\log_{10} 5^x = \log_{10} 17$$
$$x \log_{10} 5 = \log_{10} 17$$
$$x = \frac{\log_{10} 17}{\log_{10} 5} = 1.7603\ldots$$
$$\log_5 17 = 1.7603\ldots$$
$$5^{1.7603\ldots} = 17$$

◀

In this example, note that the base-5 logarithm of a number is directly proportional to the base-10 logarithm of that number. The conclusion of the example can be expressed as follows:

$$\log_5 17 = \frac{1}{\log_{10} 5} \cdot \log_{10} 17 = 1.4306\ldots \log_{10} 17$$

To find the base-5 logarithm of any number, simply multiply its base-10 logarithm by $1.4306\ldots$ (that is, divide by $\log_{10} 5$).

This proportional relationship is known as the change-of-base property. From the results of Example 3, you can write:

$$\log_5 17 = \frac{\log_{10} 17}{\log_{10} 5}$$

Notice that the logarithm with the desired base is isolated on the left side of the equation, while the two logarithms on the right side share the same base—typically one that is available on your calculator. The box below illustrates this property for bases $a$ and $b$ with argument $x$:

---

**The Change-of-Base Property of Logarithms**

$$\log_a x = \frac{\log_b x}{\log_b a} \quad \text{or} \quad \log_a x = \frac{1}{\log_b a}\left(\log_b x\right)$$

---

**Example 1.42** Find $\ln 29$ using the change-of-base property with base-10 logarithms. Check your answer directly by pressing $\ln 29$ on your calculator.

*Solution:*

$$\ln 29 = \frac{\log 29}{\log e} = \frac{1.4623\ldots}{0.4342\ldots} = 3.3672\ldots$$

$$\text{Directly:} \quad \ln 29 = 3.3672\ldots,$$

which agrees with the answer we got using the change-of-base property. ◀

> **Properties of Logarithms**
>
> The Logarithm of a Power:
> $$\log_b x^y = y \log_b x$$
>
> *Verbally*: The logarithm of a power equals the product of the exponent and the logarithm of the base. The Logarithm of a Product:
>
> $$\log_b(xy) = \log_b x + \log_b y$$
>
> *Verbally*: The logarithm of a product equals the sum of the logarithms of the factors. The Logarithm of a Quotient:
>
> $$\log_b \frac{x}{y} = \log_b x - \log_b y$$
>
> *Verbally*: The logarithm of a quotient equals the logarithm of the numerator minus the logarithm of the denominator.

## Solving Exponential and Logarithmic Equations

Logarithms provide a way to solve an equation with a variable in the exponent or to solve an equation that already contains logarithms. We will demonstrate this through the next few examples.

**Example 1.43** Solve the exponential equation $7^{3x} = 983$ algebraically, using logarithms.

*Solution:*

$$7^{3x} = 983$$
$$\log 7^{3x} = \log 983 \qquad \text{Take the base-10 logarithm of both sides.}$$
$$3x \log 7 = \log 983 \qquad \text{Apply the logarithm power property.}$$
$$x = \frac{\log 983}{3 \log 7} \qquad \text{Divide both sides by the coefficient of x.}$$
$$x = 1.1803\ldots$$

◀

**Example 1.44** Solve the equation

$$\log_2(x-1) + \log_2(x-3) = 3$$

*Solution:*

$$\log_2(x-1) + \log_2(x-3) = 3$$

$\log_2[(x-1)(x-3)] = 3$          `Apply the logarithm of a product property.`

$2^3 = (x-1)(x-3)$          `Use the definition of logarithm.`

$8 = x^2 - 4x + 3$          `Expand the product.`

$x^2 - 4x - 5 = 0$          `Reduce one side to zero. Use the symmetric`

                                        `property of equality.`

$(x-5)(x+1) = 0$          `Solve by factoring.`

$x = 5 \quad \text{or} \quad x = -1$

We need to be cautious here because the solutions in the final step are the solutions of the quadratic equation, and we must make sure they are also solutions of the original logarithmic equation. Check by substituting the solutions into the original equation.

If $x = 5$, then

$$\log_2(5-1) + \log_2(5-3)$$
$$= \log_2 4 + \log_2 2$$
$$= 2 + 1 = 3$$

If $x = -1$, then

$$\log_2(-1-1) + \log_2(-1-3)$$
$$= \log_2(-2) + \log_2(-4)$$
which is undefined.

◀

**Example 1.45** Solve the equation

$$e^{2x} - 3e^x + 2 = 0$$

*Solution:*

$$e^{2x} - 3e^x + 2 = 0$$
$$(e^x)^2 - 3e^x + 2 = 0$$

We realise that this is a quadratic equation in the variable $e^x$. Using the quadratic formula, you get

$$e^x = \frac{+3 \pm \sqrt{9 - 4(2)}}{2} = \frac{3 \pm 1}{2}$$
$$e^x = 2 \text{ or } e^x = 1$$

You now have to solve these two equations.

$$e^x = 2 \qquad\qquad e^x = 1$$
$$x = \ln 2 = 0.6931\ldots \qquad x = 0$$

Check:

$$e^{2\ln 2} - 3e^{\ln 2} + 2 \qquad\qquad (e^0)^2 - 3e^0 + 2$$
$$= \left(e^{\ln 2}\right)^2 - 3e^{\ln 2} + 2 \qquad = 1^2 - 3(1) + 2 = 0$$
$$= 2^2 - 3(2) + 2 = 0$$

Both solutions are correct.          ◀

**Example 1.46** Solve the logarithmic equation $\ln(x+3) + \ln(x+5) = 0$

*Solution:*

$$\ln(x+3) + \ln(x+5) = 0$$
$$\ln[(x+3)(x+5)] = 0$$
$$(x+3)(x+5) = e^0 = 1$$
$$x^2 + 8x + 15 = 1$$
$$x^2 + 8x + 14 = 0$$
$$x = -2.5857\ldots \quad \text{or} \quad x = -5.4142\ldots$$

Check:

$$x = -2.5857\ldots :$$
$$\ln(-2.5857\ldots + 3) + \ln(-2.5857\ldots + 5)$$
$$= \ln(0.4142\ldots) + \ln(2.4142\ldots)$$
$$= -0.8813\ldots + 0.8813\ldots = 0$$

which is ok.

$$x = -5.4142\ldots :$$
$$\ln(-5.4142\ldots + 3) + \ln(-5.4142\ldots + 5)$$
$$= \ln(-2.4142\ldots) + \ln(-0.4142\ldots)$$

which is undefined.

The only valid solution is $x = -2.5857\ldots$. ◀

# Chapter 2   Number Systems

We are so accustomed to working within the decimal system that we often forget it is a relatively recent invention and was once considered revolutionary. It is time to carefully examine how we represent numbers. Typically, we use the decimal system, where a number like 3459 is shorthand for $3 \times 1000 + 4 \times 100 + 5 \times 10 + 9$. The position of each digit is crucial, as it allows us to distinguish between values like 30 and 3. The decimal system is a **positional numeral system**, meaning it has designated positions for units, tens, hundreds, and so forth. Each digit's position implies the multiplier (a power of ten) that should be used with that digit, and each position has a value ten times that of the position to its right.

Notice that we can save space by writing 1000 as $10^3$, where the exponent 3 indicates the number of zeros. Thus, $100000 = 10^5$. If the exponent is negative, it represents a fraction, e.g., $10^{-3} = \frac{1}{1000}$. Perhaps the most ingenious aspect of the positional system was the addition of the decimal point, which allows us to include decimal fractions. For example, the number 123.456 is equivalent to:

$$1 \times 100 + 2 \times 10 + 3 \times 1 + 4 \times \frac{1}{10} + 5 \times \frac{1}{100} + 6 \times \frac{1}{1000}.$$

This can be visualised as:

| Multiplier: | ... | $10^2$ | $10^1$ | $10^0$ | . | $10^{-1}$ | $10^{-2}$ | $10^{-3}$ | ... |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Digits: | ... | 1 | 2 | 3 | . | 4 | 5 | 6 | ... |

↑
Decimal Point:

However, there is no inherent reason why we must use powers of 10, or base 10. The Babylonians, for instance, used base 60, and base 12 was very common in medieval Europe. Today, the most widely used numeral systems are summarised in Table 2.1

| Numeral system | Symbols | Base | Additional information |
| --- | --- | --- | --- |
| **Decimal** | 0-9 | 10 | - |
| **Binary** | 0, 1 | 2 | - |
| **Hexadecimal** | 0-9, A-F | 16 | $A \equiv 10, B \equiv 11, C \equiv 12, D \equiv 13, E \equiv 14, F \equiv 15$ |
| **Octal** | 0-7 | 8 | - |

**Table 2.1:** Summary of Common Numeral Systems

We begin by focusing on binary which will also receive the most detailed attention in this chapter.

## 2.1  Binary Numbers

In the binary scale, we express numbers in powers of 2 rather than the 10s of the decimal scale. For some numbers, this is easy. Recall $2^0 = 1$,

As in decimal, we write this with the position of the digit representing the power, the first place after the decimal being the $2^0$ position, the next the $2^1$, and so on. To convert a decimal number to binary, we can use the `mod` operator.

| Decimal number | | In powers of 2 | Power of 2 | | | | Binary number |
|:---:|:---:|:---:|:---:|:---:|:---:|:---:|:---:|
| | | | 3 | 2 | 1 | 0 | |
| 8 | = | $2^3$ | 1 | 0 | 0 | 0 | 1000 |
| 7 | = | $2^2 + 2^1 + 2^0$ | 0 | 1 | 1 | 1 | 111 |
| 6 | = | $2^2 + 2^1$ | 0 | 1 | 1 | 0 | 110 |
| 5 | = | $2^2 + 2^0$ | 0 | 1 | 0 | 1 | 101 |
| 4 | = | $2^2$ | 0 | 1 | 0 | 0 | 100 |
| 3 | = | $2^1 + 2^0$ | 0 | 0 | 1 | 1 | 11 |
| 2 | = | $2^1$ | 0 | 0 | 1 | 0 | 10 |
| 1 | = | $2^0$ | 0 | 0 | 0 | 1 | 1 |

**Table 2.2:** Decimal Numbers in Binary Representation

As an example, consider 88 in decimal or $88_{10}$. We would like to write it as a binary number. We take the number and successively divide `mod` 2. See below:

| Step Number $n$ | $x_n$ | $x_n/2$ | $x_n \bmod 2$ |
|:---:|:---:|:---:|:---:|
| 0 | 88 | 44 | 0 |
| 1 | 44 | 22 | 0 |
| 2 | 22 | 11 | 0 |
| 3 | 11 | 5 | 1 |
| 4 | 5 | 2 | 1 |
| 5 | 2 | 1 | 0 |
| 6 | 1 | 0 | 1 |

**Table 2.3:** Conversion of Decimal 88 to Binary

Writing the last column in reverse, that is from the bottom up, we have 1011000, which is the binary form of 88, i.e., $88_{10} = 1011000_2$.

Binary decimals are less common but quite possible. Thus, 101.1011 is just $2^2 + 2^0 + 2^{-1} + 2^{-3} + 2^{-4}$, which is, after some calculation, 5.6875. We have seen how to turn the integer part of a decimal number into a binary number, and we can do the same with a decimal fraction. Consider 0.6875. As before, we draw up a table:

| Step Number $n$ | $x_n$ | $x_n \times 2$ | $\lfloor x_n \times 2 \rfloor$ |
|:---:|:---:|:---:|:---:|
| 0 | 0.6875 | 1.375 | 1 |
| 1 | 0.375 | 0.75 | 0 |
| 2 | 0.75 | 1.5 | 1 |
| 3 | 0.5 | 1 | 1 |

**Table 2.4:** Conversion of Decimal Fraction 0.6875 to Binary

Giving, reading down, $0.6875_{10} = 1011_2$.

## Binary Expansion

The process outlined in the previous section is called **binary expansion** and refers to the representation of a number in the binary (base-2) numeral system. Every decimal number can be expressed as a sum of powers of 2, where each power corresponds to a binary digit (bit) in the number's binary form.

Let's reconsider the decimal number 88. To find its binary expansion, we identify the largest power of 2 less than or equal to 88 and continue subtracting powers of 2 until we reach 0.

First, we note that $2^6 = 64$ is the largest power of 2 less than 88:

$$88 = 64 + 24$$

Next, we find that $2^4 = 16$ is the largest power of 2 less than 24:

$$24 = 16 + 8$$

Finally, $2^3 = 8$ exactly matches the remainder:

$$8 = 8 + 0$$

Thus, we have:

$$88 = 2^6 + 2^4 + 2^3$$

In binary, each of these powers of 2 is represented by a '1' in the corresponding place value, with '0' in place values where no power of 2 contributes:

$$88_{10} = 1011000_2$$

To summarise:

- $2^6 = 64$ corresponds to the leftmost '1' in the binary expansion.
- $2^4 = 16$ corresponds to the next '1'.
- $2^3 = 8$ corresponds to the next '1'.
- The remaining digits are '0' because $2^5$, $2^2$, $2^1$, and $2^0$ do not contribute to the value 88.

Thus, the binary expansion of 88 is $1011000_2$. This method of representing numbers is fundamental in computer science and digital electronics, where binary representation is the standard for data storage and processing.

## Binary Operations

Binary operations are basic arithmetic operations performed on binary numbers. These operations are essential in computing and digital systems, as they form the foundation for how computers process and manipulate data.

Binary addition, subtraction, and multiplication are similar to their decimal counterparts but follow simpler rules due to the binary system's limited digits. For example, binary addition follows these rules:

$$0 + 0 = 0, \quad 0 + 1 = 1, \quad 1 + 0 = 1, \quad 1 + 1 = 10$$

In this case, $1+1$ results in $10_2$, which means 0 with a carry of 1 to the next higher bit. Binary subtraction and multiplication follow similar straightforward rules that are easy to implement in digital systems.

The XOR (exclusive OR) operation is another important binary operation. XOR produces a 1 if the two bits being compared are different and a 0 if they are the same:

$$0 \oplus 0 = 0, \quad 0 \oplus 1 = 1, \quad 1 \oplus 0 = 1, \quad 1 \oplus 1 = 0$$

In binary addition, the XOR operation is used to add two bits without considering any carry from a previous bit. This is because XOR effectively performs addition modulo 2, which aligns perfectly with how binary addition works. For example:

| Bit 1 | Bit 2 | XOR (Sum) | AND (Carry) |
|-------|-------|-----------|-------------|
| 0 | 0 | 0 | 0 |
| 0 | 1 | 1 | 0 |
| 1 | 0 | 1 | 0 |
| 1 | 1 | 0 | 1 |

In the case of $1+1$, XOR gives a sum of 0 and an AND operation (which detects the carry) gives a carry of 1, resulting in the binary number 10.

$$0 + 0 = 0$$
$$0 + 1 = 1$$
$$1 + 1 = 10 \quad \text{so we carry 1 and leave a zero}$$
$$1 + 1 + 1 = 1 + (1 + 1) = 1 + 10 = 11.$$

We can write this in very much the same way as for a decimal addition:

```
   | 1 | 1 | 0 | 1 | 0 | 1 |
 + | 1 | 0 | 1 | 1 | 1 | 0 |
 1 | 1 | 0 | 0 | 0 | 1 | 1 |   Sum
 ↑                 ↑
```

The right-hand arrow shows where we carry a 1. The left-hand arrow shows where we have $1 + 1 + 1$ so we carry a 1 and have a 1 left over.

As we will see below, we will often need to handle multiple carries. There are two ways to handle this which resemble the methods we know from the decimal system. We will explain using an example.

## Method 1: Column-wise Binary Addition with Multiple Carries

Consider

```
        1 1 1 1 1
    +   1 1 1 0 1
    +   1 1 1 0 1
    +   1 1 1 1 1
```

**Step 1: Add the Rightmost Column**

Start by adding the rightmost bits:

$$1 + 1 + 1 + 1 = 100_2 \quad (\text{ which is binary for } 4)$$

Reading the result from right to left (i.e. from *least significant bit* (LSB) to the *most significant bit*(MSB))

- write down the 0
- carry the 0 to the next column
- carry the 1 to the third column

You end up with

$$
\begin{array}{ccccccc}
 & & & {\scriptstyle 1} & {\scriptstyle 0} & & \\
 & & 1 & 1 & 1 & 1 & 1 \\
+ & & 1 & 1 & 1 & 0 & 1 \\
+ & & 1 & 1 & 1 & 0 & 1 \\
+ & & 1 & 1 & 1 & 1 & 1 \\
\hline
 & & & & & & 0 \\
\end{array}
$$

**Step 2: Add the Second Column from the Right**

Next, add the second column:

$$0 + 1 + 0 + 0 + 1 = 10_2 \quad (\text{ which is binary for } 2)$$

Reading the result from LSB to MSB:

- write down the 0
- carry the 1 to the next column

You end up with

$$
\begin{array}{ccccccc}
 & & & {\scriptstyle 1} & & & \\
 & & & {\scriptstyle 1} & & & \\
 & & 1 & 1 & 1 & 1 & 1 \\
+ & & 1 & 1 & 1 & 0 & 1 \\
+ & & 1 & 1 & 1 & 0 & 1 \\
+ & & 1 & 1 & 1 & 1 & 1 \\
\hline
 & & & & & 0 & 0 \\
\end{array}
$$

**Step 3: Add the Third Column from the Right**

Now, add the third column:

$$1 + 1 + 1 + 1 + 1 + 1 = 110_2 \quad (\text{ which is binary for } 6)$$

Reading the result from LSB to MSB

- write down the 0
- carry the 1 to the fourth column
- carry the 1 to the fifth column

You end up with

Our sum so far:

```
        1   1
        1 1 1 1 1
 +        1 1 1 0 1
 +        1 1 1 0 1
 +        1 1 1 1 1
          ─────────
              0 0 0
```

## Step 4: Add the Fourth Column from the Right

Move to the fourth column:

$$1 + 1 + 1 + 1 + 1 = 101_2 \quad (\text{ which is binary for } 5)$$

Reading the result from LSB to MSB

- write down the 1
- carry the 0 to the fifth column
- carry the 1 to the sixth column

You end up with

```
          0
        1   1
        1 1 1 1 1
 +        1 1 1 0 1
 +        1 1 1 0 1
 +        1 1 1 1 1
          ─────────
            1 0 0 0
```

## Step 5: Add the Leftmost Column

Add the leftmost column:

$$1 + 1 + 1 + 1 + 1 = 101_2 \quad (\text{ which is binary for } 5)$$

Reading the result from LSB to MSB

- write down the 1
- carry the 0 to the sixth column
- carry the 1 to the seventh column

This results in

```
              0
            1 1
          1 1 1 1 1
+         1 1 1 0 1
+         1 1 1 0 1
+         1 1 1 1 1
          ─────────
          1 1 0 0 0
```

**Step 6: Add the Remaining Carries**

Finally, add the remaining carries:

```
            1 1 1 1 1
+           1 1 1 0 1
+           1 1 1 0 1
+           1 1 1 1 1
        ───────────────
        1 1 1 1 0 0 0
```

The following example demonstrates the entire process by using different colors to distinguish each column and the corresponding carries they produce. Note that the last two digits in the sum are colored black, as they do not result from any specific column but are instead generated solely from the carries.

```
            1   0
            1 0     1
            1 1 1 0
          1 1 1 1 1
+         1 1 1 0 1
+         1 1 1 0 1
+         1 1 1 1 1
      ───────────────
      1 1 1 1 0 0 1
```

## Method 2: Direct Summation and Simplification

We will illustrate the second method using the same example. In the previous case, we carried the actual binary number to the next columns. In this method, we write down 0 if the sum is even and 1 if the sum is odd. Every time a sum a multiple of 2, we carry a 1 to the next columns, and then continue this process for each column, including the carries in the calculation of that column.

**Step 1: Add the Rightmost Column**

Add bits in column 1 (from counting from MSB):

$$1 + 1 + 1 + 1 = 100_2 \quad (\text{ which is binary for } 4)$$

Reading the result from LSB to MSB

- write down the 0
- carry a 1 for the first multiple of 2
- carry a 1 for the second multiple of 2

This results in

$$
\begin{array}{r}
\phantom{+}\quad 1 \\
\phantom{+}\quad 1 \\
1\ 1\ 1\ 1\ 1 \\
+\qquad 1\ 1\ 1\ 0\ 1 \\
+\qquad 1\ 1\ 1\ 0\ 1 \\
+\qquad \underline{1\ 1\ 1\ 1\ 1} \\
0
\end{array}
$$

**Step 2: Add the Second Column from the Right**

Add bits in column 2 (from counting from MSB):

$$1 + 1 + 1 + 1 = 100_2 \quad (\text{ which is binary for } 4)$$

Reading the result from LSB to MSB

- write down the 0
- carry a 1 for the first multiple of 2
- carry a 1 for the second multiple of 2

This results in

$$
\begin{array}{r}
\phantom{+}\quad 1 \\
\phantom{+}\quad 1 \\
1\ 1\ 1\ 1\ 1 \\
+\qquad 1\ 1\ 1\ 0\ 1 \\
+\qquad 1\ 1\ 1\ 0\ 1 \\
+\qquad \underline{1\ 1\ 1\ 1\ 1} \\
0\ 0
\end{array}
$$

**Step 3: Add the Third Column from the Right** Add bits in column 3 (from counting from MSB):

$$1 + 1 + 1 + 1 + 1 + 1 = 110_2 \quad (\text{ which is binary for } 6)$$

Reading the result from LSB to MSB

- write down the 0
- carry a 1 for the first multiple of 2
- carry a 1 for the second multiple of 2
- carry a 1 for the third multiple of 2

This results in

```
                  1
                  1
                  1
              1 1 1 1 1
    +           1 1 1 0 1
    +           1 1 1 0 1
    +         1 1 1 1 1
                  0 0 0
```

**Step 4: Add the Fourth Column from the Right**

Add bits in column 4 (from counting from MSB):

$$1 + 1 + 1 + 1 + 1 + 1 + 1 = 111_2 \quad (\text{ which is binary for 7})$$

Reading the result from LSB to MSB

- write down the 1
- carry a 1 for the first multiple of 2
- carry a 1 for the second multiple of 2
- carry a 1 for the third multiple of 2

This results in

```
                  1
                  1
                  1
              1 1 1 1 1
    +           1 1 1 0 1
    +           1 1 1 0 1
    +         1 1 1 1 1
                1 0 0 0
```

**Step 5: Add the Leftmost Column**

Add bits in leftmost column):

$$1 + 1 + 1 + 1 + 1 + 1 + 1 = 111_2 \quad (\text{ which is binary for 7})$$

Reading the result from LSB to MSB

- write down the result in binary, i.e. 1 1 1

This results in

$$
\begin{array}{rccccccc}
 & & 1 & 1 & 1 & 1 & 1 \\
+ & & & 1 & 1 & 1 & 0 & 1 \\
+ & & & 1 & 1 & 1 & 0 & 1 \\
+ & & & 1 & 1 & 1 & 1 & 1 \\
\hline
 & 1 & 1 & 1 & 1 & 0 & 0 & 0
\end{array}
$$

which corresponds to the result we obtained above. While not demonstrated explicitly here, subtraction works in a similar fashion.

By using one of these methods for handling multiple carries, allow us to also multiply two binary numbers.

## Multiplication in Binary

Multiplication in binary is technically easier than multiplication in decimal. In binary operations, we work exclusively with two digits: 0 and 1. This means that both the the multiplier[1] and multiplicand consist of 0's and 1' (and so does the multiplicand). The process of finding the binary product is analogous to traditional multiplication in the decimal system. The four five steps involved in multiplying binary digits are:

$$0 \times 0 = 0$$
$$0 \times 1 = 0$$
$$1 \times 0 = 0$$
$$1 \times 1 = 1$$
$$1 \times 10_2 = 10_2 \quad \text{(multiplying by base } 10_2 \text{ adds a 0 to the end)}$$

The last step means that $101_2 \times 10_2 = 1010_2$ which is analogous to the decimal case: $143_{10} \times 10_{10} = 1430_{10}$.

We will illustrate the process by supplying a couple of examples.

## Example 2.1

$$
\begin{array}{rlcccc}
 & (1) & & 1 & 0 & 0 \\
\times & (2) & & & 1 & 1 \\
\hline
 & (3) & & 1 & 0 & 0 \\
+ & (4) & 1 & 0 & 0 & 0 \\
\hline
 & (5) & 1 & 1 & 0 & 0
\end{array}
$$

Here are the steps:

- Multiply the multiplicand (line 1) by the LSB of the multiplier (line 2), which in this case is 1.
- Record this result in line 3.

---

[1]The "multiplicand" is the number that has to be multiplied, and the "multiplier" is the number by which it is multiplied.

- Append a 0 to line 4 to account for the shift to the next power of 2 in the multiplier.
- Multiply the multiplicand (line 1) by the next bit of the multiplier (line 2), which is also 1 in this case.
- Add this result to line 4, after the 0 you appended earlier.
- Finally, sum the values in lines 3 and 4, as outlined in the previous section, to obtain the final result in line 5.

We offer two additional examples:

**Example 2.2**

```
              1  0  1
×  _ _ _ _ _ _ _ 1  0  1  1
              1  0  1
+          1  0  1  0
+       0  0  0  0  0
+    1  0  1  0  0  0
     1  1  0  1  1  1
```

**Example 2.3**

```
           1  0  0  1  1  1  0
×  _ _ _ _ _ _ _ _ _ _ _ _ 1  0  1
           1  0  0  1  1  1  0
+       1  0  0  1  1  1  0  0  0
        1  1  0  0  0  0  1  1  0
```

In Example 2.3 notice that we omitted the row of zeroes that the second value of the multiplier would have produced, and notice even further that we added two 0's before restating the multiplicand in the sum.

Binary multiplication, like binary addition, is a core operation in computer arithmetic. By breaking the process down into manageable steps—multiplying individual bits and then summing the results—it becomes clear how similar it is to the multiplication methods we use in the decimal system. The main difference is the simplicity and efficiency of working within the binary system, where only the digits 0 and 1 are involved.

We notice how binary multiplication builds on binary addition. Each step, involving shifts and sums, essentially consists of repeated additions adjusted by powers of two. A strong grasp of binary addition naturally leads to a better understanding of binary multiplication and its applications.

## 2.2 Octal and Hexadecimal

Octal is a base-8 numbering system that uses the digits 0 through 7. It is closely related to binary, which is a base-2 system. The connection between the two lies in how easily binary numbers can be converted to octal and vice versa. Each octal digit corresponds to exactly three binary digits (bits), making conversions straightforward. For example, the binary number '110' converts directly to the octal digit '6'. Because of

this close relationship, octal is often used as a shorthand for binary in computing, particularly in contexts where grouping binary digits in sets of three simplifies reading and interpreting binary data.

$$12_8 = 1 \cdot 8^1 + 2 \cdot 8^0 = 10_{10}$$
$$3021_8 = 3 \cdot 8^3 + 0 \cdot 8^2 + 2 \cdot 8^1 + 1 \cdot 8^0 = 1553_{10}$$

Since $8$ is $2^3$, we can express it in binary:

$3 \rightarrow 011$

$0 \rightarrow 000$

$2 \rightarrow 010$

$1 \rightarrow 001$

Thus, $3021_8 = 011000010001_2 = 11000010001_2$

We obtain the final result by removing leading zeros.

Hexadecimal is a base-16 numbering system that uses sixteen distinct symbols: the digits 0-9 and the letters A-F, where A represents 10, B represents 11, and so on up to F, which represents 15. Hexadecimal is closely related to binary because each hexadecimal digit corresponds exactly to four binary digits (bits). This direct relationship makes it easy to convert between the two systems. For example, the hexadecimal digit 'A' translates to the binary sequence '1010'. Due to this efficiency in grouping, hexadecimal is often used in computing as a more compact and readable way to represent binary data, particularly in areas like memory addresses and colour codes in web design.

$$123_{16} = 1 \cdot 16^2 + 2 \cdot 16^1 + 3 \cdot 16^0 = 256 + 32 + 3 = 291_{10}$$
$$A2E_{16} = 10 \cdot 16^2 + 2 \cdot 16^1 + 14 \cdot 16^0 = 2560 + 32 + 14 = 2606_{10}$$

Since $16$ is $2^4$, we can, for instance, express $A2E_{16}$ in binary:

$A \rightarrow 1010$

$2 \rightarrow 0010$

$E \rightarrow 1110$

Thus, $A2E_{16} = 101000101110_2$

Similarly we get $5EB52_{16}$ as

$5 \rightarrow 0101$

$E \rightarrow 1110$

$B \rightarrow 1011$

$5 \rightarrow 0101$

$2 \rightarrow 0010$

Thus, $5EB52_{16} = 01011110101101010010_2$

Again, notice that we removed the leading 0's from $5_{16}$ when writing the result.

## 2.3 Converting Between Systems

Understanding how to convert numbers between binary, decimal, octal, and hexadecimal systems is essential in computer science and digital electronics. Each system is a different base, and each has its own applications. Here's a step-by-step guide to help you convert numbers from one system to another.

### Decimal to Binary Conversion

To convert a decimal number to binary:

1. Divide the decimal number by 2.
2. Record the remainder (it will be 0 or 1).
3. Divide the quotient by 2 and record the remainder.
4. Repeat until the quotient is 0.
5. The binary number is the sequence of remainders read from bottom to top.

**Example 2.4** Convert $23_{10}$ to binary.

*Solution:*

$$23 \div 2 = 11 \quad \text{remainder } 1$$
$$11 \div 2 = 5 \quad \text{remainder } 1$$
$$5 \div 2 = 2 \quad \text{remainder } 1$$
$$2 \div 2 = 1 \quad \text{remainder } 0$$
$$1 \div 2 = 0 \quad \text{remainder } 1$$

Thus, $23_{10} = 10111_2$. ◄

### Decimal to Octal Conversion

To convert a decimal number to octal:

1. Divide the decimal number by 8.
2. Record the remainder.
3. Divide the quotient by 8 and record the remainder.
4. Repeat until the quotient is 0.
5. The octal number is the sequence of remainders read from bottom to top.

**Example 2.5** Convert $78_{10}$ to octal.

*Solution:*

$$78 \div 8 = 9 \quad \text{remainder } 6$$
$$9 \div 8 = 1 \quad \text{remainder } 1$$
$$1 \div 8 = 0 \quad \text{remainder } 1$$

Thus, $78_{10} = 116_8$. ◄

## Decimal to Hexadecimal Conversion

To convert a decimal number to hexadecimal:

1. Divide the decimal number by 16.
2. Record the remainder (use A, B, C, D, E, F for remainders 10, 11, 12, 13, 14, 15 respectively).
3. Divide the quotient by 16 and record the remainder.
4. Repeat until the quotient is 0.
5. The hexadecimal number is the sequence of remainders read from bottom to top.

**Example 2.6** Convert $255_{10}$ to hexadecimal.

*Solution:*

$$255 \div 16 = 15 \quad \text{remainder } 15 \quad \text{(F)}$$
$$15 \div 16 = 0 \quad \text{remainder } 15 \quad \text{(F)}$$

Thus, $255_{10} = FF_{16}$. ◀

## Binary to Decimal Conversion

To convert a binary number to decimal:

1. Multiply each bit by 2 raised to the power of its position, starting from 0 on the right.
2. Sum all the products.

Notice that this amounts to the method outlined above about binary expansion.

**Example 2.7** Convert $1101_2$ to decimal.

*Solution:*

$$1 \cdot 2^3 + 1 \cdot 2^2 + 0 \cdot 2^1 + 1 \cdot 2^0 = 8 + 4 + 0 + 1 = 13_{10}$$

◀

## Binary to Octal Conversion

To convert a binary number to octal:

1. Group the binary digits into sets of three, starting from the right. Add leading zeros if necessary.
2. Convert each group of three binary digits to its octal equivalent.

**Example 2.8** Convert $110110_2$ to octal.

*Solution:*

$$110 \rightarrow 6$$
$$110 \rightarrow 6$$

Thus, $110110_2 = 66_8$. ◀

## Binary to Hexadecimal Conversion

To convert a binary number to hexadecimal:

1. Group the binary digits into sets of four, starting from the right. Add leading zeros if necessary.
2. Convert each group of four binary digits to its hexadecimal equivalent.

**Example 2.9** Convert $10110101_2$ to hexadecimal.

*Solution:*

$$1011 \rightarrow B$$
$$0101 \rightarrow 5$$

Thus, $10110101_2 = B5_{16}$. ◀

## Octal to Binary Conversion

To convert an octal number to binary: Convert each octal digit to its 3-bit binary equivalent!

**Example 2.10** Convert $57_8$ to binary.

*Solution:*

$$5 \rightarrow 101$$
$$7 \rightarrow 111$$

Thus, $57_8 = 101111_2$. ◀

## Octal to Decimal Conversion

To convert an octal number to decimal:

1. Multiply each digit by 8 raised to the power of its position, starting from 0 on the right.
2. Sum all the products.

**Example 2.11** Convert $157_8$ to decimal.

*Solution:*

$$1 \cdot 8^2 + 5 \cdot 8^1 + 7 \cdot 8^0 = 64 + 40 + 7 = 111_{10}$$

◀

## Octal to Hexadecimal Conversion

To convert an octal number to hexadecimal:

1. First, convert the octal number to binary.
2. Then, convert the binary number to hexadecimal by grouping the binary digits in sets of four.

**Example 2.12** Convert $157_8$ to hexadecimal.

*Solution:*

$$1 \to 001$$
$$5 \to 101$$
$$7 \to 111$$

Thus, $157_8 = 001101111_2 = 6F_{16}$. ◀

## Hexadecimal to Binary Conversion

To convert a hexadecimal number to binary: Convert each hexadecimal digit to its 4-bit binary equivalent.

**Example 2.13** Convert $2B_{16}$ to binary.

*Solution:*

$$2 \to 0010$$
$$B \to 1011$$

Thus, $2B_{16} = 00101011_2$. ◀

## Hexadecimal to Decimal Conversion

To convert a hexadecimal number to decimal:

1. Multiply each digit by 16 raised to the power of its position, starting from 0 on the right.
2. Sum all the products.

**Example 2.14** Convert $2B_{16}$ to decimal.

*Solution:*

$$2 \cdot 16^1 + 11 \cdot 16^0 = 32 + 11 = 43_{10}$$

◀

## Hexadecimal to Octal Conversion

To convert a hexadecimal number to octal:

1. First, convert the hexadecimal number to binary.
2. Then, convert the binary number to octal by grouping the binary digits in sets of three.

**Example 2.15** Convert $2B_16$ to octal.

*Solution:*

$$2 \to 0010$$
$$B \to 1011$$

Thus, $2B_{16} = 00101011_2 = 53_8$. ◀

**Final Thoughts on Conversion**

The concept of expansion plays a central role in these conversions. Whether you are expanding a decimal number into its binary, octal, or hexadecimal form, or converting a binary number into its octal or hexadecimal equivalent, you are more or less expressing the number in terms of powers of the base. The expansion method is essentially the same for each system as it boils down to dividing by the highest power of the base recursively:

$$7562_{10} = 1 \cdot 16^3 + 3466 = 1 \cdot 16^3 + 13 \cdot 16^2 + 138$$
$$= 1 \cdot 16^3 + 13 \cdot 16^2 + 8 \cdot 16^1 + 10 \cdot 16^0 = 1\,D\,8\,A$$

By understanding these expansions and the relationships between these number systems, you can efficiently switch between them, allowing you to represent and manipulate data in the most suitable format for any given situation.

# Chapter 3  Set Theory

In software engineering, we are constantly working with collections of things: users in a system, records in a database, or nodes in a network. Set theory provides the formal mathematical language to describe and manipulate these collections with precision and clarity. It is the bedrock upon which many core computer science concepts—from database query languages like SQL to data structures and algorithmic logic—are built.

This chapter introduces the fundamental principles of set theory. We will begin with the simple, intuitive idea of a set and explore the formal notation used to define them. We will then cover the essential relationships between sets, such as subsets, and the core operations used to combine them, including unions, intersections, and complements. These operations directly correspond to the logical operators (`OR`, `AND`, `NOT`) that govern the flow of your code. Finally, we will explore ordered collections called tuples and introduce a simple method for proving set equalities.

## 3.1  What is a Set?

A set is an unordered collection of distinct objects. The objects within a set are called its **elements** or **members**. We can think of a set as a simple container where items are grouped together, and the order in which we list them does not matter. For example, the set of primary colors can be written as $\{\text{red}, \text{yellow}, \text{blue}\}$ or equally as $\{\text{blue}, \text{red}, \text{yellow}\}$.

Sets are a cornerstone of modern mathematics and a fundamental concept in computer science. They form the logical basis for everything from database query languages and data structures to the specification of programming language types.
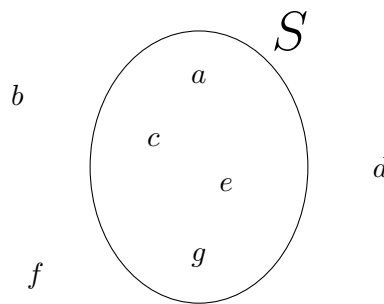


The elements of set $S$ are $a, c, e$, and $g$.

Figure 3.1: A set $S$ containing four elements. The objects $b, d$, and $f$ are not elements of $S$.

### Specifying a Set

There are two primary ways to describe a set: by explicitly listing its members or by defining a property that its members must satisfy.

**Listing Notation (Roster Method)**

The most direct way to define a set is by listing all its elements between curly braces, $\{\}$. This is known as the **roster method**.

For example:

- The set of the first five letters of the alphabet is $A = \{a, b, c, d, e\}$.
- The set of the first three positive integers is $C = \{1, 2, 3\}$.
- A set can contain different types of elements: $D = \{\text{Alice}, 42, \pi\}$.

When using the roster method, there are two fundamental rules:

1. **Order does not matter.** A set is defined only by the elements it contains, not by the sequence in which they are listed. For example, $\{1, 2, 3\}$ is the exact same set as $\{3, 1, 2\}$.
2. **Each element must be unique.** An element is either in a set or it is not. Listing an element more than once is redundant and does not change the set. For instance, the set $\{a, a, b, c, c\}$ is simply $\{a, b, c\}$.

**Set-Builder Notation**

When listing every element is impractical or impossible (for example, with infinite sets), we use **set-builder notation**. This method defines a set by stating a property or rule that its elements must satisfy. The notation uses a vertical bar '|' or a colon ':', which is read as "such that."

The general structure is:

$\{\text{variable} \mid \text{a property the variable must satisfy}\}$

For example:

- $A = \{l \mid l$ is a vowel in the English alphabet$\}$
  This is read as: "$A$ is the set of all elements $l$ such that $l$ is a vowel." This is another way of writing $A = \{a, e, i, o, u\}$.
- $C = \{n \mid n \in \mathbb{Z}$ and $0 < n < 4\}$
  This is read as: "$C$ is the set of all numbers $n$ such that $n$ is an integer and $n$ is greater than 0 and less than 4." This defines the set $\{1, 2, 3\}$.
- $E = \{x \mid x$ is an even integer$\}$
  This defines the infinite set of all even integers: $\{\ldots, -4, -2, 0, 2, 4, \ldots\}$.

Set-builder notation is extremely powerful because it allows us to define large, complex, or even infinite sets with a short and precise description.

## 3.2 Important Sets: The Number Systems

Numbers are the foundation of mathematics and computation. The different categories of numbers we use every day, from counting items to measuring continuous values, can be formally defined as sets. Understanding these fundamental sets is crucial for any software engineer, as they underpin data types, arithmetic logic, and numerical algorithms.

## The Main Number Sets

The following sets are some of the most important in mathematics and are used throughout science and engineering.

**Natural Numbers ($\mathbb{N}$)** The set of positive integers used for counting: $\{1, 2, 3, \dots\}$. Sometimes, this set is defined to include $0$. Due to this ambiguity, it is often clearer to specify *positive integers* or *non-negative integers*.

**Integers ($\mathbb{Z}$)** The set of all positive and negative whole numbers, including zero: $\{\dots, -2, -1, 0, 1, 2, \dots\}$.

**Rational Numbers ($\mathbb{Q}$)** The set of all numbers that can be expressed as a fraction $\frac{p}{q}$, where $p$ and $q$ are integers and $q \neq 0$. This includes all integers and terminating or repeating decimals. Examples: $\frac{1}{2}$, $-5$, $0.25$.

**Real Numbers ($\mathbb{R}$)** The set of all numbers on the number line. It includes both rational numbers and irrational numbers (like $\pi$ or $\sqrt{2}$), which cannot be expressed as simple fractions.

**Complex Numbers ($\mathbb{C}$)** The set of all numbers that can be expressed in the form $a + bi$, where $a$ and $b$ are real numbers and $i$ is the imaginary unit, satisfying $i^2 = -1$.

Visualizing the relationships between these sets is key to understanding them. A Venn diagram shows the hierarchy of how these sets are nested within one another, while a number line illustrates how they cover or populate the continuum of values.
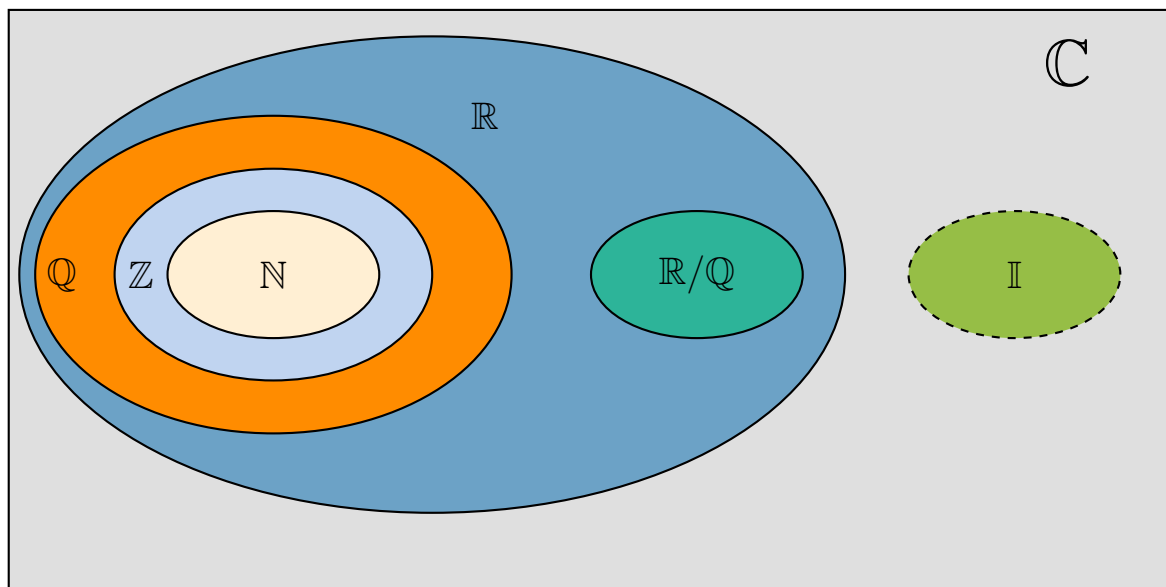


**Figure 3.2:** A Venn diagram illustrating the hierarchical relationship between the major number sets.

**Remark:** The Venn diagram in Figure 3.2 shows that the set of real numbers ($\mathbb{R}$) is composed exclusively of rational ($\mathbb{Q}$) and irrational numbers. There is no real number that is not one or the other. This is why the set of irrational numbers is formally denoted as $\mathbb{R} \setminus \mathbb{Q}$, which means "the set of all real numbers, excluding the rational numbers."

## Interval Notation

In many applications, we need to refer to a continuous range of real numbers. **Interval notation** is a convenient shorthand for describing such subsets of $\mathbb{R}$.
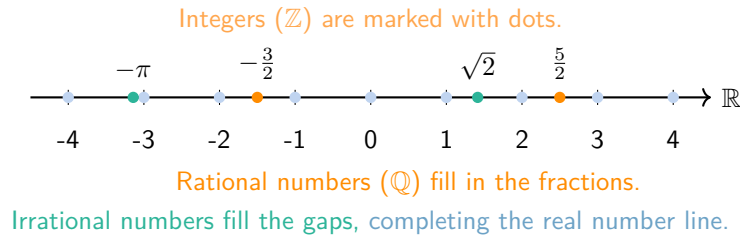
**Figure 3.3:** The real number line, populated by integers, rationals, and irrationals.

An interval is defined by its two endpoints. We use square brackets '[' ']' to indicate that an endpoint is included in the set, and parentheses '(' ')' to indicate that it is excluded.

**Closed Interval:** Includes both endpoints.
$$[a, b] = \{x \in \mathbb{R} \mid a \leq x \leq b\}$$
**Open Interval:** Excludes both endpoints.
$$(a, b) = \{x \in \mathbb{R} \mid a < x < b\}$$
**Half-Open Intervals:** Includes one endpoint but not the other.
$$[a, b) = \{x \in \mathbb{R} \mid a \leq x < b\} \quad \text{and} \quad (a, b] = \{x \in \mathbb{R} \mid a < x \leq b\}$$

**Example 3.1** Interval Notation

- $[-2, 3]$ is the set of all real numbers from $-2$ to 3, including $-2$ and 3.
- $(-2, 3)$ is the set of all real numbers between $-2$ and 3.
- $[0, 100)$ represents all numbers from 0 up to (but not including) 100.

Intervals can also be unbounded, extending towards positive or negative infinity ($\infty$). Since infinity is not a number, it is always excluded with a parenthesis.

$$(a, \infty) = \{x \in \mathbb{R} \mid x > a\} \quad \text{and} \quad (-\infty, b] = \{x \in \mathbb{R} \mid x \leq b\}$$

**Example 3.2** Unbounded Interval

- $(3, \infty)$ represents all real numbers greater than 3.
- $(-\infty, 5)$ represents all real numbers less than 5.
- $(-\infty, 0]$ represents the set of all non-positive real numbers.

## 3.3 Relationships Between Sets

Understanding a set is not just about its elements, but also how it relates to other sets. This section defines the fundamental relationships that allow us to compare and classify sets.

### Subsets and Proper Subsets

One of the most basic relationships is that of inclusion, where one set is contained within another.

> **Definition 3.1 (Subset)**
>
> A set $A$ is a **subset** of a set $B$ if every element of $A$ is also an element of $B$. We write this as $A \subseteq B$. ♣

For example, if $A = \{1, 2\}$ and $B = \{1, 2, 3\}$, then $A \subseteq B$ because every element in $A$ is also in $B$. By this definition, every set is a subset of itself (i.e., $A \subseteq A$).
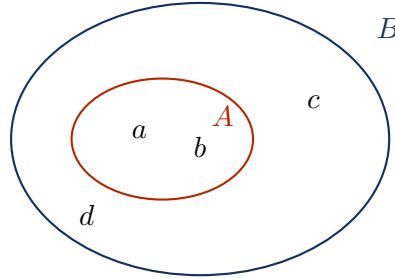


**Figure 3.4:** Set $A = \{a, b\}$ is a subset of set $B = \{a, b, c, d\}$, denoted $A \subseteq B$.

Sometimes we want to specify that a set is a subset of another but is not equal to it.

> **Definition 3.2 (Proper Subset)**
>
> A set $A$ is a **proper subset** of a set $B$ if $A \subseteq B$ and $A \neq B$. This means that $B$ must contain at least one element that is not in $A$. We write this as $A \subset B$. ♣

Using the previous example, since $B$ contains the element $3$ which is not in $A$, we can say that $A$ is a proper subset of $B$, or $A \subset B$.

## The Universal Set and the Empty Set

Two special sets act as the boundaries for set theory: the set containing everything and the set containing nothing.

> **Definition 3.3 (Universal Set)**
>
> The **universal set**, denoted by $U$, is the set of all possible elements under consideration in a given context. All other sets in that context are considered subsets of the universal set. ♣

The universal set is represented in Venn diagrams by a rectangle that encloses all other sets. For example, if we are discussing integers, the universal set would be $U = \mathbb{Z}$. If we were discussing students at a university, $U$ would be the set of all enrolled students.
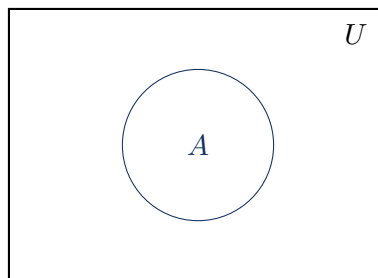


**Figure 3.5:** The universal set $U$ contains all elements and sets under consideration.

> **Definition 3.4 (Empty Set)**
>
> The **empty set** (or **null set**) is the unique set containing no elements. It is denoted by $\emptyset$ or by $\{\}$. ♣

The empty set has a crucial property:

*The empty set is a subset of every set.*

This is because there are no elements in $\emptyset$ that are not in any other set $A$. Therefore, for any set $A$, it is always true that $\emptyset \subseteq A$.
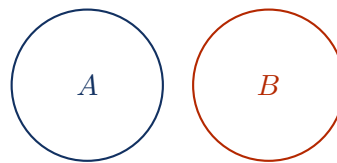
## Disjoint Sets

Sometimes, sets have no relationship at all because they are entirely separate.

> **Definition 3.5 (Disjoint Sets)**
>
> Two sets, $A$ and $B$, are **disjoint** if they have no elements in common. In other words, their intersection is the empty set: $A \cap B = \emptyset$. ♣

For example, the set of even integers $E = \{\dots, -2, 0, 2, \dots\}$ and the set of odd integers $O = \{\dots, -3, -1, 1, 3, \dots\}$ are disjoint.



Disjoint Sets

**Figure 3.6:** Sets $A$ and $B$ are disjoint because they do not overlap.

## 3.4 Properties of Sets

Beyond the relationships between sets, we can also describe their intrinsic properties. The two most fundamental properties are a set's size (its cardinality) and the collection of all its possible subsets (its power set).

## Cardinality

The most basic property of a finite set is its size.

> **Definition 3.6 (Cardinality)**
>
> The **cardinality** of a finite set $A$, denoted $|A|$, is the number of distinct elements in the set. ♣

For example:

- If $A = \{a, b, c, d\}$, then $|A| = 4$.
- If $B = \{n \in \mathbb{Z} \mid 0 < n < 5\}$, then $B = \{1, 2, 3, 4\}$ and $|B| = 4$.
- For the empty set, $|\emptyset| = 0$.

The concept of cardinality can be extended to infinite sets, but that is a more advanced topic beyond the scope of this chapter. For our purposes, cardinality is a simple count of the elements.

## Power Sets

One of the most powerful constructs in set theory is the idea of creating a set that contains all possible subsets of another set.

> **Definition 3.7 (Power Set)**
>
> The **power set** of a set $A$, denoted $\mathcal{P}(A)$, is the set of all subsets of $A$. The elements of the power set are themselves sets. ♣

The power set always includes the empty set ($\emptyset$) and the set $A$ itself.

**Example 3.3** Finding the Power Set of $A = \{1, 2, 3\}$

To find $\mathcal{P}(A)$, we list all possible subsets of $A$, grouped by their cardinality:

- Subsets of size 0: $\{\emptyset\}$
- Subsets of size 1: $\{\{1\}, \{2\}, \{3\}\}$
- Subsets of size 2: $\{\{1, 2\}, \{1, 3\}, \{2, 3\}\}$
- Subsets of size 3: $\{\{1, 2, 3\}\}$

Combining all these subsets into a single set gives us the power set:

$$\mathcal{P}(A) = \{\emptyset, \{1\}, \{2\}, \{3\}, \{1, 2\}, \{1, 3\}, \{2, 3\}, \{1, 2, 3\}\}$$

**Remark:**[Cardinality of a Power Set] For a finite set $A$ with cardinality $|A| = n$, the cardinality of its power set is $|\mathcal{P}(A)| = 2^n$.

This is a crucial formula for computer science. The reason is that for each of the $n$ elements in set $A$, we can make a binary choice: either we **include** it in a subset or we **exclude** it. With two choices for each of the $n$ elements, there are $2 \times 2 \times \cdots \times 2$ ($n$ times), or $2^n$, total possible combinations, which corresponds to the total number of possible subsets.

## 3.5 Operations on Sets

Just as we can perform arithmetic operations on numbers, we can perform operations on sets to create new sets. These operations form the foundation of set algebra. For a software engineer, the most powerful insight is that set operations are a direct parallel to the logical operations of **Boolean algebra**. Every rule you learn for sets has an equivalent rule in logic and digital circuit design.

> **The Duality of Sets and Boolean Algebra**
>
> The relationship between set theory and Boolean algebra is so direct that they are considered "dually isomorphic." This means they are structurally identical. Understanding one is understanding the other. The key translations are:
>
> | Set Theory | | Boolean Algebra / Logic |
> |---|---|---|
> | Union ($\cup$) | $\Longleftrightarrow$ | Boolean Sum ($+$) or OR |
> | Intersection ($\cap$) | $\Longleftrightarrow$ | Boolean Product ($\cdot$) or AND |
> | Complement ($A^c$) | $\Longleftrightarrow$ | Complementation ($\bar{x}$) or NOT |
> | The Universal Set ($U$) | $\Longleftrightarrow$ | True (1) |
> | The Empty Set ($\emptyset$) | $\Longleftrightarrow$ | False (0) |

We will now explore the primary set operations, keeping this duality in mind.

### Intersection

The intersection of two sets contains only the elements that are common to both sets. It corresponds to the logical AND operator.

> **Definition 3.8 (Intersection)**
>
> The **intersection** of sets $A$ and $B$, denoted $A \cap B$, is the set containing all elements that are in both $A$ and $B$.
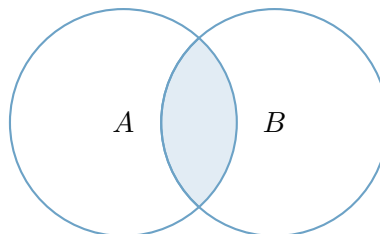> $$A \cap B = \{x \mid x \in A \text{ and } x \in B\}$$ ♣



**Figure 3.7:** The shaded region represents the intersection $A \cap B$.

### Union

The union of two sets contains all the elements that appear in either set (or both). It corresponds to the logical OR operator.

> **Definition 3.9 (Union)**
>
> The **union** of sets $A$ and $B$, denoted $A \cup B$, is the set containing all elements that are in $A$, or in $B$, or in both.
> $$A \cup B = \{x \mid x \in A \text{ or } x \in B\}$$ ♣

### Set Difference

The difference between two sets contains the elements that are in the first set but *not* in the second set.
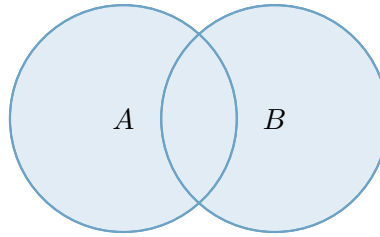
**Figure 3.8:** The shaded region represents the union $A \cup B$.

> **Definition 3.10 (Set Difference)**
>
> The **difference** of set $A$ and set $B$, denoted $A \setminus B$, is the set containing all elements that are in $A$ but not in $B$.
>
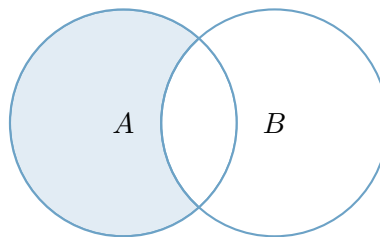> $A \setminus B = \{x \mid x \in A \text{ and } x \notin B\}$ ♣



**Figure 3.9:** The shaded region represents the difference $A \setminus B$.

## Symmetric Difference

The symmetric difference contains all elements that are in one set or the other, but not in both. It corresponds to the logical `XOR` operator.

> **Definition 3.11 (Symmetric Difference)**
>
> The **symmetric difference** of sets $A$ and $B$, denoted $A \oplus B$, is the set of elements which are in either of the sets, but not in their intersection.
>
> $A \oplus B = (A \cup B) \setminus (A \cap B)$ ♣
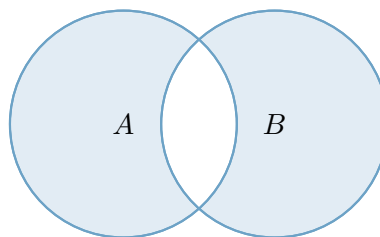


**Figure 3.10:** The shaded region represents the symmetric difference $A \oplus B$.

## Complement

The complement of a set contains all the elements in the universal set that are *not* in the set itself. It corresponds to the logical `NOT` operator.

**Definition 3.12 (Complement)**

The **complement** of a set $A$, denoted $A^c$, is the set of all elements in the universal set $U$ that are not in $A$.
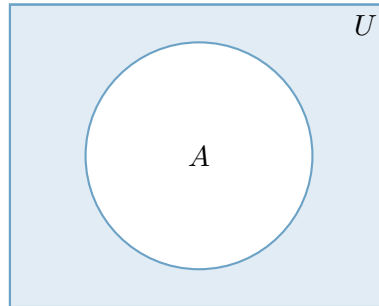
$A^c = U \setminus A$ ♣



Figure 3.11: The shaded region represents the complement $A^c$.

## 3.6 Cartesian Products and Tuples

While sets are unordered collections, we often need to work with ordered collections in computer science, such as coordinates, database records, or structured data. The Cartesian product is the set operation that allows us to create these ordered structures.

### Cartesian Product

The Cartesian product creates a new set from two or more existing sets, consisting of all possible ordered combinations of their elements.

**Definition 3.13 (Cartesian Product)**

The **Cartesian product** of sets $A$ and $B$, denoted $A \times B$, is the set of all possible ordered pairs $(a, b)$, where the first element $a$ is from $A$ and the second element $b$ is from $B$.

$A \times B = \{(a, b) \mid a \in A \text{ and } b \in B\}$ ♣

The name comes from the Cartesian coordinate system, where any point on a 2D plane can be represented by an ordered pair $(x, y)$ from the Cartesian product of the real numbers, $\mathbb{R} \times \mathbb{R}$.
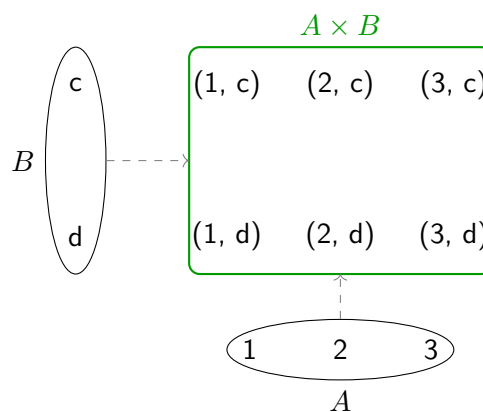


Figure 3.12: The Cartesian product of $A = \{1, 2, 3\}$ and $B = \{c, d\}$ results in a set of 6 ordered pairs.

**Remark:** The order of the sets in a Cartesian product matters. The set $A \times B$ is generally not equal to the set $B \times A$. For example, the pair $(1, c)$ is in $A \times B$, but the pair $(c, 1)$ would be in $B \times A$.

### Tuples

The elements of a Cartesian product are called **tuples**. If the product involves $n$ sets, its elements are called **n-tuples**.

- A **2-tuple**, such as $(a, b)$, is more commonly known as an **ordered pair**.
- A **3-tuple** has the form $(a, b, c)$.
- An **n-tuple** has the form $(a_1, a_2, \ldots, a_n)$.

The defining characteristic of a tuple is that **order matters**. This makes tuples fundamentally different from sets.

- For sets: $\{1, 2, 3\} = \{3, 2, 1\}$
- For tuples: $(1, 2, 3) \neq (3, 2, 1)$

This property makes tuples ideal for representing data where the position of an element carries meaning, such as a database record '(UserID, Name, Email)'.

## 3.7 Proving Set Equalities

In software development, simplifying complex conditional logic is crucial for writing efficient and readable code. Similarly, in set theory, we often need to prove that two different expressions describe the exact same set. There are two primary methods for this: using membership tables and applying set identities.

### Method 1: Membership Tables

A membership table is a tool used to prove that two set expressions are equal by checking every possible combination of an element's membership in the constituent sets.

This method is the set-theory equivalent of using a **truth table** in Boolean algebra. Instead of checking for TRUE or FALSE, we check if an element is a member (represented by a $1$) or not a member (represented by a $0$) of a set. If the columns for two different set expressions are identical in all rows, the expressions are proven to be equal.

**Example 3.4** Showing that $A \cap B = B \setminus (B \setminus A)$

To prove this equality, we construct a membership table for all combinations of membership in sets $A$ and $B$.

| $A$ | $B$ | $A \cap B$ | $B \setminus A$ | $B \setminus (B \setminus A)$ |
|-----|-----|------------|-----------------|-------------------------------|
| 1   | 1   | **1**      | 0               | **1**                         |
| 1   | 0   | **0**      | 0               | **0**                         |
| 0   | 1   | **0**      | 1               | **0**                         |
| 0   | 0   | **0**      | 0               | **0**                         |

**Table 3.1:** Fundamental Set Identities and their Boolean Algebra Counterparts.

| Identity Name | Set Identity | Boolean Identity |
|---|---|---|
| Identity Laws | $A \cup \emptyset = A$ | $x + 0 = x$ |
| | $A \cap U = A$ | $x \cdot 1 = x$ |
| Domination Laws | $A \cup U = U$ | $x + 1 = 1$ |
| | $A \cap \emptyset = \emptyset$ | $x \cdot 0 = 0$ |
| Idempotent Laws | $A \cup A = A$ | $x + x = x$ |
| | $A \cap A = A$ | $x \cdot x = x$ |
| Complement Laws | $A \cup A^c = U$ | $x + \overline{x} = 1$ |
| | $A \cap A^c = \emptyset$ | $x \cdot \overline{x} = 0$ |
| Commutative Laws | $A \cup B = B \cup A$ | $x + y = y + x$ |
| | $A \cap B = B \cap A$ | $x \cdot y = y \cdot x$ |
| Associative Laws | $(A \cup B) \cup C = A \cup (B \cup C)$ | $(x + y) + z = x + (y + z)$ |
| | $(A \cap B) \cap C = A \cap (B \cap C)$ | $(x \cdot y) \cdot z = x \cdot (y \cdot z)$ |
| Distributive Laws | $A \cap (B \cup C) = (A \cap B) \cup (A \cap C)$ | $x \cdot (y + z) = (x \cdot y) + (x \cdot z)$ |
| | $A \cup (B \cap C) = (A \cup B) \cap (A \cup C)$ | $x + (y \cdot z) = (x + y) \cdot (x + z)$ |
| De Morgan's Laws | $(A \cap B)^c = A^c \cup B^c$ | $\overline{x \cdot y} = \overline{x} + \overline{y}$ |
| | $(A \cup B)^c = A^c \cap B^c$ | $\overline{x + y} = \overline{x} \cdot \overline{y}$ |
| Absorption Laws | $A \cup (A \cap B) = A$ | $x + (x \cdot y) = x$ |
| | $A \cap (A \cup B) = A$ | $x \cdot (x + y) = x$ |

**Conclusion:** Since the column for $A \cap B$ is identical to the column for $B \setminus (B \setminus A)$ for all possible membership combinations, the two expressions are equal.

## Method 2: Set Identities

While membership tables are effective, they can become very large as the number of sets increases. A more algebraic approach is to use **set identities** to simplify one expression until it matches the other. These identities are fundamental laws that govern how set operations behave.

The table below shows the most important set identities. Notice the striking similarity to the laws of Boolean algebra—they are structurally identical.

**Example 3.5** Proving $A \cup (B \cap A^c) = A \cup B$ using identities

$$
\begin{aligned}
A \cup (B \cap A^c) &= (A \cup B) \cap (A \cup A^c) && \text{by Distributive Law} \\
&= (A \cup B) \cap U && \text{by Complement Law} \\
&= A \cup B && \text{by Identity Law}
\end{aligned}
$$

## 3.8 Computer Representation of Sets

While set theory provides the abstract language for collections, computer science requires concrete and efficient ways to implement these ideas. For finite universal sets, one of the most elegant and performant methods is to represent sets using **bit strings**. This technique requires two conditions:

1. The universal set $U$ must be finite.
2. The elements of $U$ must have a fixed, agreed-upon order.

Let $U = \{a_1, a_2, \ldots, a_n\}$. Any subset $A \subseteq U$ can be represented by a bit string of length $n$, where the $i$-th bit is 1 if $a_i \in A$, and 0 if $a_i \notin A$.

**Example 3.6** Representing Sets as Bit Strings

Let the universal set be $U = \{1, 2, 3, 4, 5, 6, 7, 8\}$.

- The set $A = \{1, 3, 4, 8\}$ is represented by the bit string 10110001.
- The set $B = \{2, 3, 8\}$ is represented by the bit string 01100001.
- The set of all even numbers, $\{2, 4, 6, 8\}$, is 01010101.
- The empty set, $\emptyset$, is 00000000.

The true power of this representation is that set operations map directly to extremely fast, low-level bitwise operations that processors can execute in a single cycle.

---

**Set Operations as Bitwise Operations**

Let the bit strings for sets $A$ and $B$ be $s_A$ and $s_B$.
- **Union ($A \cup B$)** corresponds to a bitwise `OR` operation.
- **Intersection ($A \cap B$)** corresponds to a bitwise `AND` operation.
- **Complement ($A^c$)** corresponds to a bitwise `NOT` (one's complement) operation.
- **Symmetric Difference ($A \oplus B$)** corresponds to a bitwise `XOR` operation.

---

**Example 3.7** Performing Set Operations with Bit Strings

Using $U = \{1, 2, 3, 4, 5, 6, 7, 8\}$, let $A = \{1, 3, 4, 8\}$ and $B = \{2, 3, 8\}$.

|  | Set Representation | Bit String Representation |
|---|---|---|
| Set $A$ | $\{1, 3, 4, 8\}$ | 10110001 |
| Set $B$ | $\{2, 3, 8\}$ | 01100001 |
| $A \cup B$ | $\{1, 2, 3, 4, 8\}$ | 10110001 OR 01100001 = 11110001 |
| $A \cap B$ | $\{3, 8\}$ | 10110001 AND 01100001 = 00100001 |
| $A^c$ | $\{2, 5, 6, 7\}$ | NOT 10110001 = 01001110 |

This bit string representation is fundamental in many areas of computing, including file permissions in operating systems, database indexing, network protocols, and graphics programming, as it provides a way to manage and query collections with maximum efficiency.

# Chapter 4   Combinatorics and Probability Theory

Imagine you are tasked with forming teams of 3 for a semester project in a class of 45 students. Initially, the order in which you choose the team members does not matter, so you are just concerned with combinations. The number of ways to form a team of 3 from 45 students comes out to 14,190 possibilities!

The following semester introduces the students to Scrum project management, where each team must have three specific roles: Scrum Master, Product Owner, and Development Team. This small change, specifying roles, suddenly transforms the problem from a simple *combination* into a *permutation*. Now, the number of possible ways to assign these roles leaps to 85,140!

Frustrated by the sheer number of options, the 45 students throw a party to relax. Being well-mannered, they decide that everyone should shake hands with every other person exactly once. After a few minutes, they calculate the total number of handshakes — 990. The students are once again surprised by how something as simple as shaking hands can add up so quickly.

As the night progresses, one student proposes a fun game — a random drawing for five door prizes, each unique. With 45 students in attendance and only five prizes available, the chance of winning nothing becomes a concerning 89 per cent. The students quickly realize that the odds are not in their favor.

Not ready to give up on their luck, a smaller group decides to flip a coin 10 times, with the hope of landing exactly five heads to win the game. However, when they learn that the probability of this happening is only about 25 per cent, their spirits dampen further.

The students conclude that rather than relying on chance, it is time to dive deeper into understanding combinatorics and probability theory. Armed with this knowledge, they can better predict outcomes and avoid future disappointments at both parties and project planning.

## 4.1  Sample Space and Events

A **random experiment** is one that can lead to different outcomes, even when repeated under the same conditions. This randomness is a fundamental aspect of many engineering tasks.

Think of it this way: Let us say you are testing the speed of a website under different conditions. Sometimes it loads quickly, and other times it is slower. Even if you are using the same code and server, things like network traffic or server load make the results vary each time.

> **Definition 4.1 (Random Experiment)**
>
> A random experiment is one that can give different results, even if you do everything the same each time. ♣

Or, imagine you are measuring the signal strength in a wireless device. You might get slightly different readings each time because of things like interference or small changes in the environment.

This randomness shows up all over the place, from software performance tests to electrical engineering experiments. It is important to expect it and include it in your thinking. Otherwise, you might make

decisions based on incomplete or misleading data. When you account for random variation, you can make smarter predictions and designs.

To model and analyze a random experiment, it is crucial to understand the set of possible outcomes that can occur. In probability theory, this set is called the **sample space**, denoted by $S$. A sample space can be either **discrete** (consisting of a finite or countably infinite set of outcomes) or **continuous** (containing an interval of real numbers). The exact definition of a sample space often depends on the objectives of the analysis.

An **outcome** is a single possible result of the random experiment, and an **event** is any subset of the sample space, which may consist of one or more outcomes. Below are some examples to illustrate these concepts:

### Example 4.1 Network Latency

Consider an experiment where you measure the latency of data packets in a network. The sample space can be defined based on the type of measurements:

- If latency is measured as a positive real number, the sample space is continuous:
  $$S = \{x \mid x > 0\}.$$
- If it is known that latency ranges between 10 and 100 milliseconds, the sample space can be refined to:
  $$S = \{x \mid 10 \leq x \leq 100\}.$$
- If the objective is to categorize latency as low, medium, or high, the sample space becomes discrete:
  $$S = \{\text{low}, \text{medium}, \text{high}\}.$$
- For a simple evaluation of whether the latency meets a standard threshold, the sample space can be reduced to:
  $$S = \{\text{pass}, \text{fail}\}.$$

Each outcome in these sample spaces represents a single possible latency measurement, and events can be defined as sets of outcomes, such as "latency is high."

Understanding the nature of sample spaces, outcomes, and events is fundamental in probability, as it allows us to define and work with probabilities of complex scenarios in various engineering contexts. Let us summarise these key concepts:

> **Definition 4.2 (Sample Spaces, Outcomes, and Events)**
>
> **Sample Space:** The set of all possible outcomes of a random experiment is called the sample space, denoted by $S$. Outcomes can be discrete or continuous, depending on the nature of the experiment.
> **Outcome:** A single possible result of a random experiment.
> **Event:** Any subset of the sample space, which may consist of one or more outcomes. ♣

### Example 4.2 Software Release Testing

Imagine a software testing process where each test case can either pass or fail. The sample space for a single test case is discrete and can be represented as:

$$S = \{\text{pass}, \text{fail}\}.$$

If you run three test cases, the combined sample space for all possible outcomes is:

$$S = \{(\text{pass, pass, pass}), (\text{pass, pass, fail}), (\text{pass, fail, pass}),$$
$$(\text{pass, fail, fail}), (\text{fail, pass, pass}), (\text{fail, pass, fail}),$$
$$(\text{fail, fail, pass}), (\text{fail, fail, fail})\}.$$

This sample space includes all sequences of outcomes for the three tests.

- The total number of possible outcomes is $2^3 = 8$.
- An event could be defined as "at least one test fails," which would include outcomes like (fail, pass, pass), (pass, fail, fail), and others where at least one test fails.

**Example 4.3** Component Quality in Manufacturing

A company manufactures electronic components, and each component is tested for compliance with quality standards. The test can return one of three outcomes: `pass`, `marginal`, or `fail`. The sample space is:

$$S = \{\text{pass, marginal, fail}\}.$$

**Event:** Suppose we are interested in the event that a component does not pass the quality test. This event is a set of outcomes:

$$E = \{\text{marginal, fail}\}.$$

This means that all the operations we have defined on sets translate directly into operations on events:

- $A \cup B$: the event that at least one of $A$ or $B$ occurs.
- $A \cap B$: the event that both $A$ and $B$ occur together.
- $\overline{A}$ or $A^c$: the event that $A$ does not occur.
- $A \setminus B$ or $A - B$: the event that $A$ occurs but $B$ does not.
- $A \triangle B$ or $A \oplus B$: the event that either $A$ or $B$ occurs, but not both (symmetric difference).

Thus, probability theory builds directly on set theory: probability assigns a numerical measure to these subsets of the sample space. In the next chapter, we will see how this measure is defined and used to reason about uncertainty.

We can summarise the considerations of this section in the following table:

| Operation | Boolean Algebra | Logic | Set Theory |
|---|---|---|---|
| NOT | $\overline{x}$ | $\neg x$ | $A^c$ or $A'$ |
| OR | $+$ | $\vee$ | $\cup$ |
| AND | $\cdot$ | $\wedge$ | $\cap$ |
| NAND | $\overline{x \cdot y}$ | $\neg(x \wedge y)$ | $(A \cap B)^c$ or $\overline{A \cap B}$ |
| NOR | $\overline{x + y}$ | $\neg(x \vee y)$ | $(A \cup B)^c$ or $\overline{A \cup B}$ |
| XOR (Symmetric Difference) | $x \oplus y$ | $(x \wedge \neg y) \vee (\neg x \wedge y)$ | $A \triangle B$ or $A \oplus B$ |
| Difference | $x \cdot \overline{y}$ | $x \wedge \neg y$ | $A - B$ or $A \setminus B$ |

**Table 4.1:** Comparison of Operators in Boolean Algebra, Logic, and Set Theory

In this book, we will use the notation $A^c$ to denote the complement of the set $A$, which includes all elements not in $A$.

## 4.2 Counting Principles

In many problems across mathematics, computer science, and engineering, determining the number of ways certain events can occur is important. Whether you're arranging elements, selecting groups, or navigating through complex scenarios, counting techniques provide the foundational tools to solve these problems. These techniques go beyond simple arithmetic and allow us to tackle questions like:

- How many ways can we arrange a set of objects?
- In how many different paths can a process unfold?
- What is the probability of a specific event occurring given multiple possibilities?

Counting techniques, such as permutations, combinations, and the multiplication rule, help us quantify these possibilities systematically.

### Multiplication Rule

We start out by discussing the most basic counting principle: the **multiplication rule**:

---
**Theorem 4.1 (Multiplication Rule)**

Let an operation be described as a sequence of $k$ steps. Assume the following conditions:
- There are $n_1$ ways to complete step 1.
- There are $n_2$ ways to complete step 2 for each way of completing step 1.
- There are $n_3$ ways to complete step 3 for each way of completing step 2, and so on.

Then, the total number of ways to complete the entire operation is given by:

$$n_1 \times n_2 \times \cdots \times n_k.$$

---

**Example 4.4** Suppose you are choosing a meal at a restaurant. You have the following options:

- 3 choices for the main course.
- 4 choices for the side dish.
- 2 choices for the drink.

Using the multiplication rule, the total number of ways to choose a meal is:

$$3 \times 4 \times 2 = 24.$$

Therefore, there are 24 different meal combinations available.

**Example 4.5** Automobile Options

An automobile manufacturer provides vehicles equipped with selected options. Each vehicle is ordered

- With or without an automatic transmission
- With or without a sunroof
- With one of three choices of a stereo system

- With one of four exterior colors

If the sample space consists of the set of all possible vehicle types, what is the number of outcomes in the sample space?

**Solution:** Using the multiplication rule, we can calculate the total number of possible vehicle types by multiplying the number of choices for each option:

- 2 choices for the transmission (with or without automatic transmission)
- 2 choices for the sunroof (with or without sunroof)
- 3 choices for the stereo system
- 4 choices for the exterior color

Therefore, the total number of possible vehicle types is:

$$2 \times 2 \times 3 \times 4 = 48$$

So, there are 48 different possible vehicle types in the sample space. ◀

## Replacement and Order in Counting

Next we turn to an important distinction between with and without replacement in counting principles:

> **Definition 4.3 (Counting with and without Replacement)**
>
> When counting the number of ways to select objects from a set, two common scenarios are:
> - **With Replacement**: An object can be selected more than once.
> - **Without Replacement**: Once an object is selected, it cannot be chosen again. ♣

**Example 4.6** Suppose you have a bag containing 5 different colored balls. You draw 2 balls:

- **With Replacement**: The first ball is placed back in the bag before drawing the second. There are $5 \times 5 = 25$ possible outcomes.
- **Without Replacement**: The first ball is not placed back, so the number of outcomes is $5 \times 4 = 20$.

**Example 4.7** Three people are drawing cards one after another from a standard deck of 52 cards. The goal is to find the Ace of Spades. Let's examine the two scenarios: with replacement and without replacement.

**Without Replacement**

In this scenario, each card drawn is not put back into the deck, reducing the total number of cards available after each draw.

- First Draw: The first person has a $\frac{1}{52}$ chance of drawing the Ace of Spades.
- Second Draw: If the first person does not draw the Ace of Spades, there are now 51 cards left, and the second person has a $\frac{1}{51}$ chance of drawing the Ace of Spades.
- Third Draw: If the Ace of Spades has not been drawn by the first two people, the third person has a $\frac{1}{50}$ chance of drawing it.

The probabilities change with each draw because the total number of cards decreases, and previously drawn cards are not available.

**With Replacement**

In this scenario, each card drawn is returned to the deck and reshuffled before the next person draws. This keeps the total number of cards constant.

- First Draw: The first person has a $\frac{1}{52}$ chance of drawing the Ace of Spades.
- Second Draw: Since the card is replaced and shuffled back into the deck, the second person also has a $\frac{1}{52}$ chance of drawing the Ace of Spades.
- Third Draw: Similarly, the third person has a $\frac{1}{52}$ chance of drawing the Ace of Spades.

The probabilities remain the same for each draw because the deck is reset to its original state after each draw.

**Example 4.8** Imagine you have a group of 5 students: Alice, Bob, Charlie, David, and Eve. You need to select 2 of them for different scenarios, illustrating when the order of selection matters and when it does not.

- **Order Matters**: Selecting Alice as Captain and Bob as Assistant Captain is a different outcome than selecting Bob as Captain and Alice as Assistant Captain.

Now, imagine you are simply selecting 2 students to form a study group with no specific roles assigned. Here, the order does not matter.

- **Order does not matters:** Choosing Alice and Bob is considered the same outcome as choosing Bob and Alice; there is no distinction between the two orders since there are no assigned roles.

Example 4.8 illustrates the distinction between *permutations* and *combinations*, two fundamental counting principles that are widely used in probability theory and combinatorics.

> **Definition 4.4 (Permutation and Combination)**
>
> - **Permutation (order matters)**: Different sequences are counted as distinct outcomes, leading to a higher count.
> - **Combination (order does not matter)**: Sequences are treated as identical, resulting in a lower count. ♣

We will first discuss permutations, which are used when the order of selection matters.

## Permutations

Consider a set of elements, such as $S = \{a, b, c\}$. A permutation of the elements is an ordered sequence of the elements. For example, $abc, acb, bac, bca, cab,$ and $cba$ are all of the permutations of the elements of $S$.

> **Proposition 4.1 (Permutations of $n$ Distinct Objects)**
>
> The number of ordered arrangements (permutations) of $n$ distinct objects is
> $$n! = n \cdot (n-1) \cdot \cdots \cdot 2 \cdot 1.$$
> ♠

This outcome is a direct application of the multiplication rule. To form a permutation, you start by choosing an element for the first position from the total of $n$ elements. Next, you choose an element for the second position from the remaining $n-1$ elements, then for the third position from the remaining $n-2$ elements, and continue this way until all positions are filled. Such arrangements are often called linear permutations.

**Example 4.9** It is said that any shuffling of a deck of card has only happened once in history. This is because the number of ways to shuffle a deck of 52 cards is 52!, which is an astronomically large number.

$$52! \approx 8.07 \times 10^{67}$$

**Example 4.10** Suppose you have 5 different books on a shelf. You want to rearrange them in a different order. The number of ways to rearrange the books is

$$5! = 5 \times 4 \times 3 \times 2 \times 1 = 120$$

There are cases where we are only interested in arranging a subset of elements from a larger set. The formula for counting these arrangements also derives from the multiplication rule.

---

**Theorem 4.2 (Permutations of Subsets)**

For integers $n \geq r \geq 0$, the number of ordered selections of $r$ distinct objects from $n$ distinct objects is

$$P_r^n = P(n, r) = n \cdot (n - 1) \cdots (n - r + 1) = \frac{n!}{(n - r)!}.$$

♡

---

**Example 4.11** Suppose you have 5 different books on a shelf, and you want to rearrange 3 of them in a different order. The number of ways to rearrange the 3 books is

$$P_3^5 = 5 \times 4 \times 3 = 60$$

**Example 4.12** There are 10 entries in a contest. Only three will win, $1^{\text{st}}$, $2^{\text{nd}}$, or $3^{\text{rd}}$ prize. What are the possible results?

*Solution:* The number of ways to award the prizes is the number of permutations of 3 objects selected from 10, which is

$$P_3^{10} = \frac{10!}{(10 - 3)!} = \frac{10!}{7!} = 10 \times 9 \times 8 = 720$$

Therefore, there are 720 possible outcomes for awarding the prizes. ◀

## Combinations

When the order of selection does not matter, we use the concept of combinations. Combinations are used when we are interested in selecting a subset of elements from a larger set without regard to the order in which they are selected. Let is start out with a couple of examples to illustrate the concept of combinations.

**Example 4.13** Suppose you have a group of 5 students: Alice, Bob, Charlie, David, and Eve. You need to select 2 of them to form a study group. The order in which you select the students does not matter. The possible combinations are:

- Alice and Bob
- Alice and Charlie
- Alice and David
- Alice and Eve

- Bob and Charlie
- Bob and David
- Bob and Eve
- Charlie and David
- Charlie and Eve
- David and Eve

The order of the students in the study group does not matter, so the combinations are considered identical.

**Example 4.14** Maria has three tickets for a concert. She'd like to use one of the tickets herself. She could then offer the other two tickets to any of four friends (Ann, Beth, Chris, Dave). How many ways can 2 people be selected from 4 to go to a concert?

**Example 4.15** A circuit board has four different locations in which a component can be placed. If three identical components are to be placed on the board, how many different designs are possible?

***Solution:*** Since you can only place one component in each slot, placing a component in any slot immediately restricts the choices for the next component.

1. Fill slots 1, 2, and 3.
2. Fill slots 1, 2, and 4.
3. Fill slots 1, 3, and 4.
4. Fill slots 2, 3, and 4.

◀

These examples illustrate the concept of combinations, where the order of selection does not matter. The formula for combinations is derived from the permutation formula by dividing out the number of ways to arrange the $r$ elements.

---
**Theorem 4.3 (Combinations)**

The number of combinations of $r$ elements selected from a set of $n$ different elements is given by
$$C_r^n = \binom{n}{r} = \frac{n!}{r!(n-r)!}$$

---

This is also sometimes referred to as the **binomial coefficient**, denoted by $\binom{n}{r}$, which is read as "n choose r". It is called the binomial coefficient because it appears in the binomial theorem, which expands the powers of a binomial expression:

---
**Theorem 4.4 (Binomial Theorem)**

In algebra, the binomial coefficient is used to expand powers of binomials. According to the binomial theorem

$$(a+b)^n = \sum_{k=0}^{n} \binom{n}{k} a^k b^{n-k}$$

---

The theorem states that the expansion of the binomial expression $(a+b)^n$ is the sum of the terms $\binom{n}{k}a^k b^{n-k}$

for $k = 0, 1, 2, \ldots, n$. The binomial coefficient $\binom{n}{k}$ gives the number of ways to choose $k$ elements from a set of $n$ elements. We will place no more emphasis on the binomial theorem here, but it is a fundamental concept in algebra and combinatorics, and is widely used in probability theory.

We will conclude our discussion of counting principles with principles of counting with replacement.

---

**Proposition 4.2 (With replacement)**

For selections from $n$ types:

- **Ordered with replacement**: $n^r$ outcomes.
- **Unordered with replacement**: $\binom{n + r - 1}{r}$ outcomes.

♠

---

## 4.3 Basic Probability

Probability quantifies the likelihood or chance that an outcome of a random experiment will occur. For instance, when you hear, "The chance of rain today is 30%," it expresses our belief about the likelihood of rain. Probabilities are numbers assigned to outcomes, ranging from 0 to 1 (or equivalently, from 0% to 100%). A probability of 0 means the outcome will not happen, while a probability of 1 means it will happen for sure.

Probabilities can be interpreted in different ways:

- **Objective (or Classical) Probability**: Often referred to as classical probability, this approach is used when outcomes are equally likely, such as in rolling a fair die or flipping a coin. Probabilities are assigned based on the assumption that each outcome has an equal chance of occurring. For example, when rolling a fair six-sided die, the probability of rolling a 3 is $\frac{1}{6}$ because there are 6 equally likely outcomes (1, 2, 3, 4, 5, 6), and only one of them is a 3. The probability is the same for all observers.
- **Relative Frequency (Empirical Probability)**: Empirical probability is based on observations from experiments rather than theoretical calculations. For example, if a software tester runs a stress test on a server 100 times and it crashes 7 times, the empirical probability of a crash is $\frac{7}{100} = 0.07$. This approach relies on actual data rather than assumptions or intuition.
- **Subjective Probability**: This reflects our personal belief or degree of confidence in an outcome. Different people might assign different probabilities to the same event based on their knowledge or perspective. You and your friends discuss Denmark's chances of winning the World Cup. Based on recent performance and team strength, you estimate a 10% chance. However, a more optimistic friend assigns a 20% chance, while another gives only 5%, considering stronger competitors. This illustrates subjective probability, where each person's estimate varies based on personal beliefs and biases rather than objective data.

When assigning probabilities, it's essential that the sum of all probabilities in an experiment equals 1, ensuring consistency with the relative frequency interpretation.

We start by establishing the Axioms of Probability, which lay the foundation for how probabilities are assigned to events. These axioms define the basic properties that every probability measure must satisfy.

> **Axiom 4.1 (Axioms of Probability)**
> - **Axiom 1:** For any event $A$, $0 \leq P(A) \leq 1$.
> - **Axiom 2:** Probability of the sample space $S$ is $P(S) = 1$.
> - **Axiom 3:** If $A_1, A_2, A_3, \cdots$ are disjoint events, then $P(A_1 \cup A_2 \cup A_3 \cdots) = P(A_1) + P(A_2) + P(A_3) + \cdots$

The property that $0 \leq P(A) \leq 1$ is equivalent to the requirement that a relative frequency must be between 0 and 1. The property that $P(S) = 1$ is a consequence of the fact that an outcome from the sample space occurs on every trial of an experiment. Consequently, the relative frequency of $S$ is 1. Property 3 implies that if the events $A_1$ and $A_2$ have no outcomes in common, the relative frequency of outcomes in $A_1 \cup A_2$ is the sum of the relative frequencies of the outcomes in $A_1$ and $A_2$.

In the next sections we will see more about the probability of events and how to calculate them.

## Probability of an Event

The probability of an event is a measure of the likelihood that the event will occur. It is denoted by $P(A)$, where $A$ is the event. The probability of an event ranges from 0 to 1, where 0 indicates that the event will not occur, and 1 indicates that the event will occur for sure.

> **Definition 4.5 (Probability of an Event)**
> The probability of an event $A$, denoted by $P(A)$, is the likelihood that event $A$ will occur. It is defined as the ratio of the number of favorable outcomes to the total number of outcomes in the sample space.
> $$P(A) = \frac{\text{Number of favorable outcomes}}{\text{Total number of outcomes}}$$

**Example 4.16** Suppose you are testing a software module with 10 different test cases. Out of these, 3 test cases are known to fail due to a bug. If you randomly select one test case to run, what is the probability that the selected test case will fail?

**Solution:** Here, the event $A$ is "the test case fails."

- Number of favorable outcomes (failing test cases) $= 3$
- Total number of outcomes (total test cases) $= 10$

Using the formula:

$$P(A) = \frac{\text{Number of favorable outcomes}}{\text{Total number of outcomes}} = \frac{3}{10} = 0.3$$

Therefore, the probability that a randomly selected test case will fail is 0.3 , indicating that there is a $30\%$ chance of failure.

◀

**Example 4.17** Imagine a software development environment where you have 50 files, consisting of 20 Python

scripts, 15 Java files, and 15 configuration files. If you randomly select one file to edit, what is the probability that the file is a Python script?

***Solution:*** Here, the event $A$ is "the selected file is a Python script."

- Number of favorable outcomes (Python scripts) $= 20$
- Total number of outcomes (total files) $= 50$

Using the formula:

$$P(A) = \frac{\text{Number of favorable outcomes}}{\text{Total number of outcomes}} = \frac{20}{50} = 0.4$$

Thus, the probability of selecting a Python script is 0.4 , meaning there is a $40\%$ chance of choosing a Python file from the set.

◀

## 4.4 Probability of Joint Events and Set Operations

Joint events are formed by applying basic set operations to individual events. Commonly, we encounter unions of events, such as $A \cup B$; intersections of events, such as $A \cap B$; and complements of events, such as $A^c$. These combined events are often of particular interest, and their probabilities can frequently be derived from the probabilities of the individual events that compose them. Understanding these set operations is essential for accurately calculating the probability of joint events. In this section, we will explore how unions of events and other set operations can be used to determine the probabilities of more complex events.

When dealing with events, the intersection represents AND while the union represents OR The probability of the intersection of events $A$ and $B$, denoted as $P(A \cap B)$, can also be expressed as $P(A, B)$ or $P(AB)$.

From the axioms of probability, we can derive the following rules of probabilities:

---

**Theorem 4.5 (Rules of Probability)**

- **Complement Rule:** The probability of the complement of event $A$ is
  $$P(A^c) = 1 - P(A)$$
- **Empty Set Rule:** The probability of the empty set is 0, i.e.,
  $$P(\emptyset) = 0$$
- **Addition Rule:** For any two events $A$ and $B$, the probability of the union of events $A$ and $B$ is given by
  $$P(A \cup B) = P(A) + P(B) - P(A \cap B)$$
- **Difference Rule:** The probability of the difference between events $A$ and $B$ is given by
  $$P(A - B) = P(A) - P(A \cap B)$$
- **Subset Rule:** If $A$ is a subset of $B$ $(A \subset B)$, then
  $$P(A) \leq P(B)$$

.                                                                            ♡

---

We can obtain the Complement Rule by noting:

$$1 = P(S) \qquad \text{(axiom 2)}$$
$$= P(A \cup A^c) \qquad \text{(definition of complement)}$$
$$= P(A) + P(A^c) \qquad \text{(since } A \text{ and } A^c \text{ are disjoint)}$$

Since $\emptyset = S^c$, we can apply part the Complement Rule to deduce that $P(\emptyset) = 1 - P(S) = 0$. This is intuitive because, by definition, an event occurs when the outcome of the random experiment is part of that event. However, since the empty set contains no elements, no outcome of the experiment can ever belong to it, making its probability zero.

The Difference Rule can be obtained by showing that $P(A) = P(A \cap B) + P(A - B)$. Note that the two sets $A \cap B$ and $A - B$ are disjoint and their union is $A$. Thus, by the third axiom of probability

$$P(A) = P((A \cap B) \cup (A - B)) \qquad \text{(since } A = (A \cap B) \cup (A - B))$$
$$= P(A \cap B) + P(A - B) \qquad \text{(since } A \cap B \text{ and } A - B \text{ are disjoint)}$$

The Addition Rule we obtain by noting that $A$ and $B - A$ are disjoint sets and their union is $A \cup B$. Thus,

$$P(A \cup B) = P(A \cup (B - A)) \qquad \text{(since } A \cup B = A \cup (B - A))$$
$$= P(A) + P(B - A) \qquad \text{(since } A \text{ and } B - A \text{ are disjoint)}$$
$$= P(A) + P(B) - P(A \cap B) \qquad \text{(by part the Difference Rule)}$$

And finally the Subset Rule is a direct consequence of the fact that if $A \subset B$, then $B$ can be written as the union of $A$ and $B - A$. Since $A$ and $B - A$ are disjoint, we have $P(B) = P(A) + P(B - A) \geq P(A)$.

We conclude this section with a few examples illustrating the application of these rules to calculate probabilities of joint events.

**Example 4.18** A company has bid on two large construction projects. The company president believes that the probability of winning the first contract is 0.6, the probability of winning the second contract is 0.4, and the probability of winning both contracts is 0.2.

  (a) What is the probability that the company wins at least one contract?
  (b) What is the probability that the company wins the first contract but not the second contract?
  (c) What is the probability that the company wins neither contract?
  (d) What is the probability that the company wins exactly one contract?

*Solution:*   Let $A$ be the event that the company wins the first contract, and $B$ be the event that the company wins the second contract. Given:

  • $P(A) = 0.6$
  • $P(B) = 0.4$
  • $P(A \cap B) = 0.2$

  (a) The probability that the company wins at least one contract is the probability of the union of events $A$ and $B$. Using the Addition Rule:

$$P(A \cup B) = P(A) + P(B) - P(A \cap B)$$
$$= 0.6 + 0.4 - 0.2$$
$$= 0.8$$

Therefore, the probability that the company wins at least one contract is 0.8.

(b) The probability that the company wins the first contract but not the second contract is the probability of the difference between events $A$ and $B$. Using the Difference Rule:

$$P(A - B) = P(A) - P(A \cap B)$$
$$= 0.6 - 0.2$$
$$= 0.4$$

Therefore, the probability that the company wins the first contract but not the second contract is 0.4.

(c) The probability that the company wins neither contract is the probability of the complement of the union of events $A$ and $B$. Using the Complement Rule:

$$P((A \cup B)^c) = 1 - P(A \cup B)$$
$$= 1 - 0.8$$
$$= 0.2$$

Therefore, the probability that the company wins neither contract is 0.2.

(d) The probability that the company wins exactly one contract is the probability of the difference between the union of events $A$ and $B$ and the intersection of events $A$ and $B$. Using the Difference Rule:

$$P((A \cup B) - (A \cap B)) = P(A \cup B) - P(A \cap B)$$
$$= 0.8 - 0.2$$
$$= 0.6$$

So, the probability that the company wins exactly one contract is 0.6.

◄

### Example 4.19

- There is a 60 percent chance that it will rain today.
- There is a 50 percent chance that it will rain tomorrow.
- There is a 30 percent chance that it does not rain either day.

(a) The probability that it will rain today or tomorrow
(b) The probability that it will rain today and tomorrow.
(c) The probability that it will rain today but not tomorrow.
(d) The probability that it either will rain today or tomorrow, but not both.

*Solution:*

(a) Let $A$ be the event that it rains today, and $B$ be the event that it rains tomorrow.

Given:

$$P(A \cup B) = 1 - P\left((A \cup B)^c\right) \qquad \text{by the Complement Rule}$$
$$= 1 - P\left(A^c \cap B^c\right) \qquad \text{by De Morgan's Law}$$
$$= 1 - 0.3$$
$$= 0.7$$

Therefore, the probability that it will rain today or tomorrow is $0.7$.

(b) The probability that it will rain today and tomorrow: this is $P(A \cap B)$. To find this we note that

$$P(A \cap B) = P(A) + P(B) - P(A \cup B)$$
$$= 0.6 + 0.5 - 0.7$$
$$= 0.4$$

(c) The probability that it will rain today but not tomorrow: this is $P\left(A \cap B^c\right)$.

$$P(A \cap B^c) = P(A - B)$$
$$= P(A) - P(A \cap B)$$
$$= 0.6 - 0.4$$
$$= 0.2$$

(d) The probability that it either will rain today or tomorrow but not both: this is $P(A-B) + P(B-A)$. We have already found $P(A - B) = .2$. Similarly, we can find $P(B - A)$:

$$P(B - A) = P(B) - P(B \cap A)$$
$$= 0.5 - 0.4$$
$$= 0.1$$

Thus,

$$P(A - B) + P(B - A) = 0.2 + 0.1$$
$$= 0.3$$

◄

# Bibliography

Cormen, T., Leiserson, C., Rivest, R., & Stein, C. (2022). *Introduction to algorithms* (4th). MIT Press.

Lay, D. (2003). *Linear algebra and its applications*. Pearson Education.

Montgomery, D. (2013). *Applied statistics and probability for engineers, 6th edition*. John Wiley; Sons, Incorporated.

Pishro-Nik, H. (2014). *Introduction to probability, statistics, and random processes*. Kappa Research, LLC.

Rosen, K. H. (2012). *Discrete mathematics and its applications* (7th). McGraw-Hill Education.

# Appendix A: Important Concepts

This appendix is a collection of important mathematical concepts that are frequently used in software engineering. The content of this appendix is based on the concepts in this book.

---

**Proposition A.1 (Order of Operations)**

To evaluate mathematical expressions, operations are performed in the following order:

1. **Brackets (Parentheses):** First, perform all operations inside brackets or parentheses.
2. **Exponents and Radicals:** Next, evaluate exponents (powers) and radicals (roots).
3. **Multiplication and Division:** Then, perform multiplication and division from left to right.
4. **Addition and Subtraction:** Finally, execute addition and subtraction from left to right. ♠

---

**Proposition A.2 (Rules for Calculations with Fractions)**

For $a, b, c, m \in \mathbb{R}$, with $a, b, c, m \neq 0$ where required, the following identities hold:

(1) $\dfrac{a}{b} \times m = \dfrac{am}{b}$

(2) $\dfrac{a}{b} \div m = \dfrac{a}{bm}$

(3) $m \div \dfrac{a}{b} = \dfrac{mb}{a}$

(4) $\dfrac{a}{b} \times \dfrac{c}{a} = \dfrac{c}{b}$

(5) $\dfrac{a}{b} \div \dfrac{c}{a} = \dfrac{a^2}{bc}$

(6) $\dfrac{a}{b} = \dfrac{ac}{bc}$

(7) $\dfrac{a}{b} + \dfrac{c}{a} = \dfrac{a^2 + bc}{ab}$ ♠

---

**Proposition A.3 (Properties of Integer Exponents)**

Let $n, m \in \mathbb{Z}$. Then the following hold (with $x, y \in \mathbb{R}$ and nonzero where stated):

(1) $x^n \cdot x^m = x^{n+m}$,

(2) $\dfrac{x^n}{x^m} = x^{n-m}$ with $x \neq 0$,

(3) $x^n \cdot y^n = (xy)^n$,

(4) $\dfrac{x^n}{y^n} = \left(\dfrac{x}{y}\right)^n$ with $y \neq 0$,

(5) $\left(x^n\right)^m = x^{nm}$,

(6) $x^1 = x$. ♠

**Proposition A.4 (More Properties of Integer Exponents)**

Let $n, m \in \mathbb{Z}$. Then the following hold (with $x, y \in \mathbb{R}$ and nonzero where stated):

(7) $\quad x^0 = 1 \qquad\qquad x \neq 0$

(8) $\quad \dfrac{1}{x^m} = x^{-m} \qquad x \neq 0$

♠

**Rules for rearranging formulae**

The following operations can be performed on both sides of the formula:

- Add the same quantity to both sides
- Subtract the same quantity from both sides
- Multiply both sides by the same quantity - remember to multiply all terms
- Divide both sides by the same quantity - remember to divide all terms
- Apply a function to both sides, such as squaring or finding the reciprocal

**Definition A.1 (Injective and Surjective Functions)**

A function $f : A \to B$ is called **one-to-one** (or **injective**) if different elements in $A$ map to different elements in $B$. A function $f : A \to B$ is called **onto** (or **surjective**) if every element in $B$ is the image of at least one element in $A$.

♣

**Definition A.2 (Inverse Functions)**

Let $f$ be a one-to-one correspondence from the set $A$ to the set $B$. The inverse function of $f$ is the function that assigns to an element $b$ belonging to $B$ the unique element $a$ in $A$ such that $f(a) = b$. The inverse function of $f$ is denoted by $f^{-1}$. Hence, $f^{-1}(b) = a$ when $f(a) = b$.
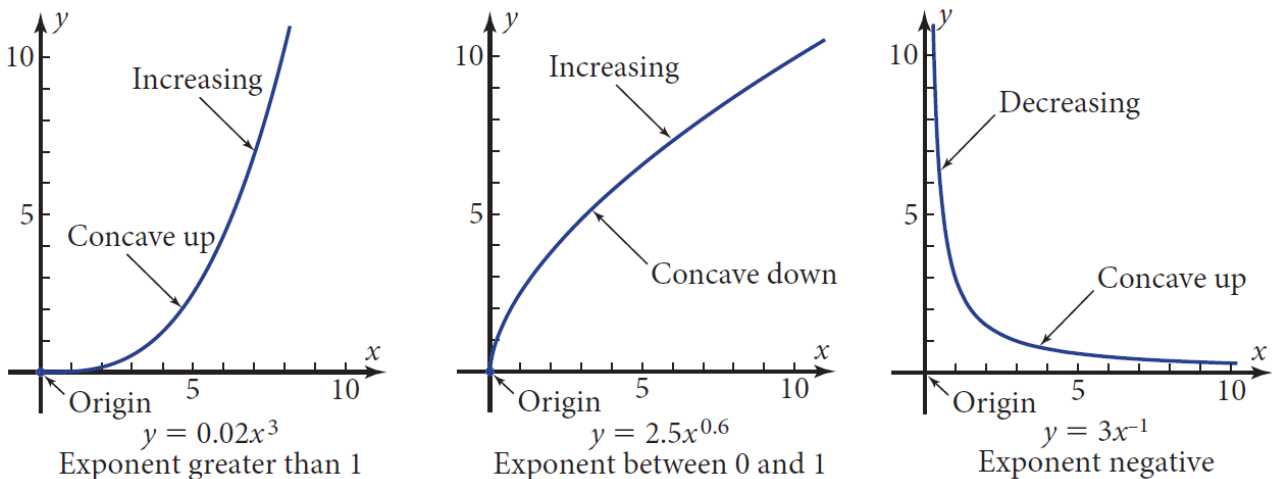
♣



**Figure A.1:** Power functions

**Definition A.3 (Base-10 Logarithms)**

$$\log x = y \iff 10^y = x$$

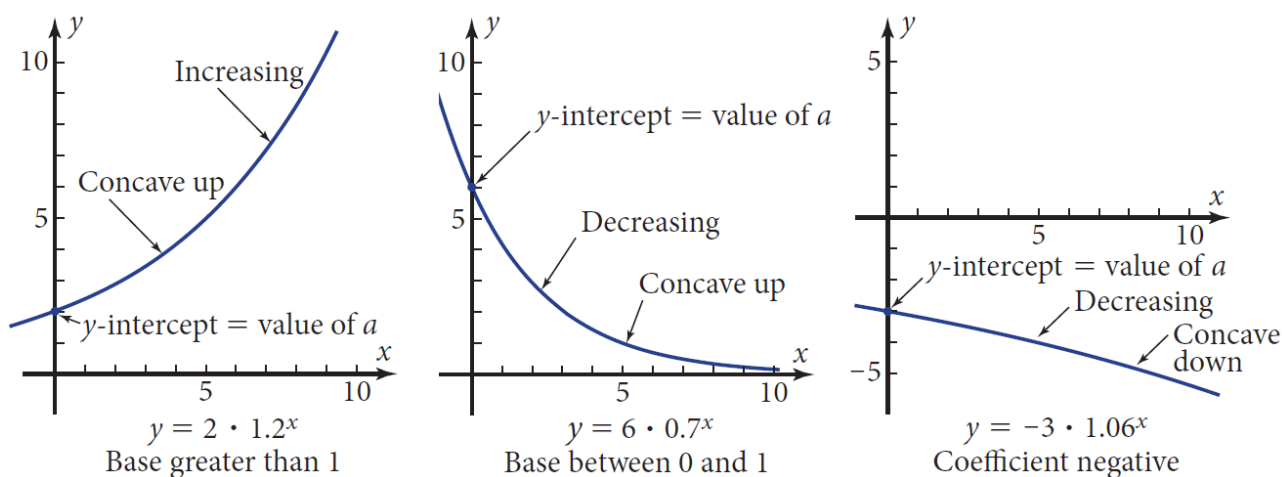*Verbally*: $\log x$ is the exponent in the power of 10 that gives $x$

♣

Figure A.2: Exponential functions

## Properties of base-10 logarithms

- Log of a Product:
$$\log xy = \log x + \log y$$
*Verbally*: The $\log$ of a product equals the sum of the logs of the factors.

- Log of a Quotient:
$$\log \frac{x}{y} = \log x - \log y$$
*Verbally*: The $\log$ of a quotient equals the log of the numerator minus the $\log$ of the denominator.

- Log of a Power:
$$\log x^y = y \log x$$
*Verbally*: The $\log$ of a power equals the exponent times the log of the base.

## Definition A.4 (Common Logarithm and Natural Logarithm)

*Common*: The symbol $\log x$ means $\log_{10} x$.

*Natural*: The symbol $\ln x$ means $\log_e x$, where $e$ is a constant equal to $2.71828182845\ldots$ ♣

## The Change-of-Base Property of Logarithms

$$\log_a x = \frac{\log_b x}{\log_b a} \quad \text{or} \quad \log_a x = \frac{1}{\log_b a}(\log_b x)$$

## Properties of Logarithms

The Logarithm of a Power:
$$\log_b x^y = y \log_b x$$

The Logarithm of a Product:
$$\log_b(xy) = \log_b x + \log_b y$$

The Logarithm of a Quotient:
$$\log_b \frac{x}{y} = \log_b x - \log_b y$$